

Documentación Practica Prod/Consum

1. Cambios realizados

- Los cambios que he realizado en esta práctica son los siguientes:

a) He importado la clase monitor:

```
import monitor.* ;
```

b) He hecho que la clase Buffer herede de la clase AbstractMonitor:

```
extends AbstractMonitor
```

c) He creado dos variables de la clase Condition para establecer la sincronización de cuando produce y cuando se consume:

```
private Condition produce= makeCondition();
```

```
private Condition consume = makeCondition();
```

d) En los métodos de depositar y extraer he usado las funciones acorde con las especificaciones del problema productor consumidor pero de forma que se corresponda con el uso de las utilidades de la clase AbstractMonitor:

```
public void depositar( double valor ) throws InterruptedException
{
    enter(); < ---- Implementa la exclusión mutua
    while( cont == numSlots ) produce.await(); < - Condición
de sincronización
    buffer[cont] = valor; cont++;
    consume.signal();
    leave(); < ---- Implementa la exclusión mutua
}

public double extraer() throws InterruptedException
{ double valor;
    enter();< ---- Implementa la exclusión mut
    while( cont == 0 ) consume.await() ; - Condición de
sincronización
    cont--; valor = buffer[cont] ;
    produce.signal();
    leave();< ---- Implementa la exclusión mut
    return valor;
}
```

- Salida del programa para 2 productores, 2 consumidores, en un buffer de tamaño 3 y 5 iteraciones por hebra:

```
java EjemploProductorConsumidor 2 2 3 10 10
```

```
productor 0, produciendo 0.0
productor 1, produciendo 100.0
consumidor 0, consumiendo 0.0
productor 1, produciendo 101.0
productor 0, produciendo 1.0
productor 0, produciendo 2.0
productor 0, produciendo 3.0
productor 0, produciendo 4.0
consumidor 1, consumiendo 100.0
productor 1, produciendo 102.0
consumidor 0, consumiendo 101.0
consumidor 1, consumiendo 3.0
productor 1, produciendo 103.0
productor 1, produciendo 104.0
productor 0, produciendo 5.0
consumidor 1, consumiendo 102.0
productor 1, produciendo 105.0
consumidor 0, consumiendo 4.0
consumidor 1, consumiendo 103.0
productor 0, produciendo 6.0
consumidor 1, consumiendo 5.0
productor 1, produciendo 106.0
consumidor 0, consumiendo 104.0
consumidor 1, consumiendo 105.0
productor 1, produciendo 107.0
productor 1, produciendo 108.0
productor 0, produciendo 7.0
consumidor 1, consumiendo 106.0
consumidor 0, consumiendo 6.0
consumidor 1, consumiendo 107.0
productor 1, produciendo 109.0
productor 0, produciendo 8.0
consumidor 1, consumiendo 7.0
productor 0, produciendo 9.0
consumidor 0, consumiendo 108.0
```

consumidor 1, consumiendo 109.0
consumidor 0, consumiendo 8.0
consumidor 0, consumiendo 9.0
consumidor 0, consumiendo 2.0
consumidor 0, consumiendo 1.0