

## Übung 8 [24]: Semaphore

Schreiben Sie eine Klasse Semaphore mit folgendem Interface (Verhalten wie in der Vorlesung Betriebssysteme I beschrieben):

```
public class Semaphore
{
    public Semaphore(long init);
    public void SemInit(long init);    // Initialisierung
    public void SemWait() throws InterruptedException;
    public void SemSignal();
    public long SemValue();    // Derzeitiger Stand
}
```

Wo notwendig, fügen Sie "synchronized" ein.

Zum Test schreiben Sie das im BS I-Skriptum beschriebene Programm eines Ringpuffers, für das 3 Semaphore benötigt werden. Ein Thread erzeugt Werte und ein anderer liest diese wieder aus (Übergeben Sie dem Konstruktor einen booleschen Wert, der entscheidet, welchen Part dieser Thread übernimmt). Stellen Sie durch geeignete sleep()-Anweisungen sicher, dass sowohl der Producer-Thread wie auch der Consumer-Thread einmal schneller als der andere arbeiten.