

## Übung 8 [6+18]: 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 ein Programm, welches einen Ringbuffer (Siehe Betriebssysteme 1) von Strings verwendet. Der Buffer wird von einem eigenen Thread gefüllt, der eine HTML-Datei einliest und zeilenweise in den Buffer stellt. Ein anderer Thread liest die Zeilen aus dem Buffer aus, entfernt alle HTML-Tags und gibt das Ergebnis auf der Konsole aus. HTML-Tags werden entfernt, indem alles zwischen spitzen Klammern, diese inklusive, entfernt wird (Vereinfachungen: Auch innerhalb von Strings soll dies gelten; keine Schachtelung).

**Stellen Sie durch geeignete sleep()-Anweisungen sicher, das sowohl der Producer-Thread wie auch der Consumer-Thread einmal schneller als der andere arbeiten!**