

Arbeitsblatt Semaphore

Ein Semaphore wird verwendet, um eine beschränkte Anzahl von Ressourcen zu kontrollieren. Ein Semaphore verwaltet einen Zähler, welcher mit der Methode *acquire* dekrementiert und mit der Methode *release* inkrementiert werden kann. Ist der Zähler des Semaphors 0, wenn *acquire* aufgerufen wird, so wartet dieser Aufruf, bis der Zähler wieder positiv ist. Der Zähler des Semaphors kann mit dem Konstruktor initialisiert und mit der Methode *available* abgefragt werden.

Im Folgenden ist ein Gerüst für die Klasse Semaphore gegeben. Ergänzen Sie in diesem Code die Methoden *acquire* und *release* (vielleicht sind auch noch andere Ergänzungen nötig).

```
public final class Semaphore {
    private int value;

    public Semaphore(int initial) {
        if (initial < 0) throw new IllegalArgumentException();
        value = initial;
    }

    public int available() {
        return value;
    }

    public void acquire () {

    }

    public void release() {

    }
}
```

Nachdem Sie sich die Lösung auf Papier überlegt haben, können Sie vom AD das Wochenprojekt 03_Condition_Synchronization.zip in ihre IDE laden. Die relevanten Klassen sind im worksheet Package zu finden. In der Klasse SemaphoreTest finden Sie noch einige Tests, die Ihre Implementierung überprüfen. Wie Sie aber wissen, können solche Tests nur Fehler aufzeigen, jedoch keine Korrektheit garantieren.