

## Assignment 9

### Deadline - Woche 14

**Achtung!** Die Aufgaben aus diesem Assignment sind *optional*!  
Die maximale Note, die erreicht werden kann, beträgt 10  
(unabhängig von der Woche)

#### Aufgabe 1

Schreiben Sie zwei Programme (ein Verbraucher- und ein Lieferungsprogramm), die jeweils ein Lebensmittellager versorgen. Das Lebensmittellager ist eine Datei, die Aufzeichnungen der folgenden Form enthält: (char Waren [20], int Menge). Mehrere Instanzen für Verbraucherprozesse und mehrere Instanzen für Lieferprozesse werden gleichzeitig auf das Lager einwirken. Simulieren Sie die Arbeit mit dem Lebensmittellager und achten Sie darauf, den Zugriff auf das Lebensmittellager zu synchronisieren, indem Sie die Datei sperren.

#### Aufgabe 2

Implementieren Sie eine Liste über eine Datei. Die Liste enthält ganze Zahlen. Das Hinzufügen eines Elements erfolgt am Anfang der Liste und das Löschen erfolgt am Ende der Liste (dh beim Hinzufügen wird eine ganze Zahl am Anfang der Datei eingefügt, und beim Entfernen wird die Datei abgeschnitten und die letzte ganze Zahl aus der Datei gelöscht). Der Zugriff auf die Datei, in der die Liste gespeichert ist, wird synchronisiert. Simulieren Sie die Listenverarbeitung, indem Sie mehrere gleichzeitige Prozesse erstellen, die Hinzufügen/Löschen-Vorgängen auf der Liste ausführen.

### Aufgabe 3

Implementieren Sie einen Stapel (stack) über eine Datei. Die Liste enthält ganze Zahlen. Das Hinzufügen eines Elements erfolgt am Anfang der Liste und das Löschen erfolgt auch am Anfang der Liste (dh beim Hinzufügen wird eine ganze Zahl am Anfang der Datei eingefügt, und beim Entfernen wird die erste Zahl aus der Datei gelöscht). Der Zugriff auf die Datei, in der der Stapel gespeichert ist, wird synchronisiert. Simulieren Sie die Stapelverarbeitung, indem Sie mehrere gleichzeitige Prozesse erstellen, die Hinzufügen/Löschen-Vorgängen auf dem Stapel ausführen.

### Aufgabe 4

Schreiben Sie zwei Programme, die dieselbe in einer Datei gespeicherte ganze Zahl (ohne Vorzeichen) inkrementieren oder dekrementieren möchten. Die Zahl hat einen Anfangswert von 500. Die Prozesse enden, wenn der Wert der Zahl entweder 0 oder 1000 ist. Am Ende jedes Prozesses wird die Anzahl der durchgeführten Operationen (Inkremente oder Dekremente) angezeigt. Der Zugriff auf die Datei erfolgt durch vorheriges Sperren. Um die Richtigkeit der Programme zu überprüfen, führen Sie Folgendes aus:

- a) mehrere Prozesse, die dekrementieren und weniger, die inkrementieren;
- b) mehrere Prozesse, die inkrementieren und weniger, die dekrementieren.

## Aufgabe 5

Schreiben Sie zwei Programme, die mit einer Datei arbeiten, die  $n^2$  ganzen Zahlen (mit Wert 0 oder 1) enthält. Die Datensätze werden als Elemente einer quadratischen Matrix  $M$  der Ordnung  $n$  betrachtet. Das erste Programm erzeugt ein zufälliges  $i$  und  $j$ ,  $i, j \in [1 \dots n]$  und prüft, ob die Anzahl der 0-Elemente um  $M[i, j]$  (seiner 3, 5 oder 8 Nachbarn) größer als der 1-Elemente ist (oder falls das gleiche Anzahl ist). Wenn ja, werden alle Nachbarn von  $M[i, j]$  0. Das zweite Programm führt den umgekehrten Vorgang aus. Jedes Programm sperrt einen geeigneten Teil der Datei, bevor es überprüft und geändert wird, und entsperrt sie anschließend. Achtung, in der Datei wird die Matrix als Vektor in einer einzelnen Zeile gespeichert. Die Prozesse enden, wenn die Matrix entweder nur 0 oder nur 1 enthält.