

## Übungen zur Vorlesung Einführung in die Informatik III

### Aufgabe 32      E/A-Kanal-Endemeldung

Im Begleitbuch zur Vorlesung werden zwei einfache Betriebssysteme für Multiplexbetrieb angegeben. Das zweite Betriebssystem arbeitet im *Zeitscheibenbetrieb*, d.h. es wechselt das zu rechnende Programm, wenn es nicht vorher zum Ende kam, nach immer gleichen Zeitabständen. Die Prozessorzuteilung kann aber auch durch Signale von peripheren Geräten gesteuert werden. Diese Signale bewirken *Unterbrechungen* der Ausführung von Benutzerprogrammen. Dadurch soll erreicht werden, dass stets genügend Benutzerprogramme im System gehalten werden und möglichst alle Geräte ständig ausgelastet sind.

- a) Stellen Sie das für den Zeitscheibenbetrieb angegebene Betriebssystem auf den Unterbrechungsbetrieb um, so dass E/A-Kanal-Endemeldungen als Unterbrechungen behandelt werden.
- b) Untersuchen Sie das unten aufgeführte Beispiel und geben Sie für jeden der drei Prozesse reader, printer und processor in Form eines Zeitstrahles an, wann er mit welchem Programm aktiv bzw. wann er idle ist. Die Ausführungszeit der Betriebssystemprozeduren sei dabei vernachlässigbar klein. Notieren Sie, welche Programme jeweils in den Warteschlangen sq, pq und cq gehalten werden.

Wie auf dem letzten Übungsblatt soll die Abarbeitung an folgendem Beispiel untersucht werden:

- Im Externspeicher liegen drei Programmstapel  $P_1$ ,  $P_2$  und  $P_3$ ;
- das Einlesen eines Programmstapels dauert jeweils 7 Zeiteinheiten;
- das Rechnen der Programme  $P_1$ ,  $P_2$ ,  $P_3$  dauert 17, bzw. 1, bzw. 4 Zeiteinheiten;
- der Ausdruck eines Programmresultates dauert jeweils 5 Zeiteinheiten;

### Aufgabe 33      Bankier-Algorithmus

Auf einem Betriebssystem, das den *Bankier-Algorithmus* zur Vermeidung von Verklemmungen bei der Betriebsmittelvergabe einsetzt, existieren 5 Betriebsmittelklassen. Es seien 4 Prozesse im System, deren maximale Anforderungen durch die Matrix

$$Max = \begin{pmatrix} 3 & 5 & 8 & 10 & 1 \\ 2 & 5 & 3 & 3 & 2 \\ 4 & 12 & 4 & 9 & 2 \\ 6 & 1 & 4 & 5 & 5 \end{pmatrix}$$

beschrieben wird. Es seien jeweils 6, 15, 8, 10, 9 Instanzen der einzelnen Betriebsmittelklassen vorhanden.

- (a) Kann das System in den Zustand gelangen, der durch die Matrix

$$Allocation = \begin{pmatrix} 0 & 2 & 1 & 1 & 1 \\ 0 & 5 & 3 & 1 & 1 \\ 0 & 7 & 1 & 2 & 1 \\ 3 & 1 & 1 & 1 & 0 \end{pmatrix}$$

beschrieben ist?

- (b) Das System befinde sich in dem Zustand

$$Allocation = \begin{pmatrix} 0 & 1 & 3 & 1 & 1 \\ 0 & 3 & 1 & 1 & 2 \\ 2 & 5 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 2 \end{pmatrix}$$

Wie muss sich das Betriebssystem verhalten, wenn nun Prozess 3 eine Instanz des 3. Betriebsmittels anfordert?

- (c) Angenommen, das Betriebssystem vergäbe in dem Zustand *Allocation* aus Aufgabe (b) auf Anforderung dem Prozess 4 eine weitere Instanz des 4. Betriebsmittels. Kommt es dann im System zwingend zu einem Deadlock? Welche Fälle können eintreten?

### **Aufgabe 34 Bankier-Algorithmus in Java**

Ziel dieser Aufgabe ist die Implementierung und die exemplarische Erprobung einer verklemmungsfreien Betriebsmittelzuteilung mit Hilfe des Bankier-Algorithmus.

- a) Implementieren Sie die Nutzer der Betriebsmittel in der Klasse `Prozess` als Unterklasse von `Thread`. Über den Konstruktor von `Prozess` sollen einer Instanz eines Prozesses eine eindeutige Identifikation (z.B. Nummer des Prozesses) sowie der Vektor der maximalen Betriebsmittelforderungen zugewiesen werden.

Jeder Prozess fordert nach seinem Start parallel zu allen anderen Prozessen und in zufälliger Reihenfolge sukzessive seine Betriebsmittel bis zur Maximalforderung an, und gibt am Ende alle Betriebsmittel zurück. Nach jeder Anforderung oder Freigabe von Betriebsmittel soll jeder Prozess, seinem Namen und die von ihm belegten Betriebsmittel auf dem Bildschirm ausgeben.

- P -

- b) Implementieren Sie die Klasse `Bankier`, mit den Methoden

- `public synchronized void belegen(int Kunde, int BMKlasse, int Menge)` und
- `public synchronized void freigeben(int Kunde, int BMKlasse, int Menge),`

die für eine sichere (bzgl. Verklemmungen) Zuteilung der Betriebsmittel sorgt. Mit dem Konstruktor von `Bankier` soll dem Bankier die Matrix der maximalen Betriebsmittelanforderungen der Prozesse mitgeteilt werden.

- c) Nutzen Sie `Bankier` und `Prozess` zur Simulation der verklemmungsfreien Zuteilung von Krediten durch eine Bank und 5 Kunden. D.h. es gibt nur eine Betriebsmittelklasse: Geld.

- d)** Nutzen Sie `Bankier` und `Prozess` zur Simulation des Beispiels mit 5 Betriebsmittelklassen und 4 Prozessen aus Aufgabe 33.
- e)** Reichern Sie Ihre Programme mit Bildschirmausgaben an, die den Ablauf des Bankieralgorithmus nachvollziehbar werden lassen, und senden sie die Ausgaben Ihrer Testprogramme an Ihren Tutor.