

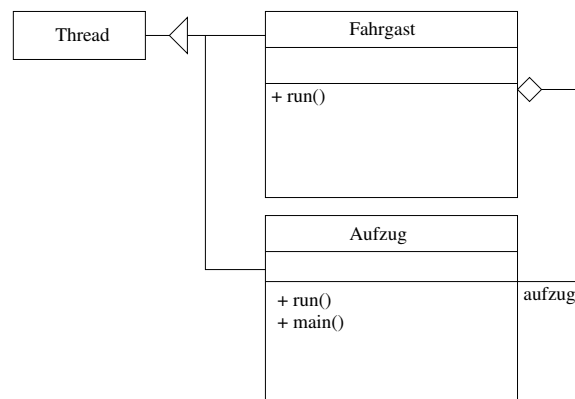
Übungen zu Einführung in die Informatik II

Aufgabe 19 Pater-Noster

Ein Pater-Noster ist ein Aufzug der ständig zwischen dem untersten und obersten Stockwerk hin und her pendelt. Die Fahrgäste können dabei den Aufzug nicht explizit anfordern, sondern müssen warten, bis der Aufzug vorbeikommt; dann können sie einsteigen.

Ziel dieser Aufgabe ist es wesentliche Abläufe eines solchen Aufzugs zu simulieren. Hierzu werden die von Java zur Verfügung gestellten Parallelisierungstechniken benötigt.

Die Grundstruktur der Simulation sei durch folgendes Klassenmodell gegeben:



a) Vervollständigen Sie das Klassenmodell, indem sie in den beiden Klassen Aufzug und Fahrgast Methoden angeben, die

- das Weiterfahren des Aufzugs,
- das kurze Anhalten des Fahrstuhls in einem Stockwerk,
- das Warten des Fahrgastes auf den Aufzug,
- das Ein- und Aussteigen des Fahrgastes

simulieren.

b) Implementieren Sie nun die beiden Klassen Aufzug bzw. Fahrgast und testen Sie ihr Programm.

Aufgabe 20 Speisende Philosophen

Fünf Philosophen sitzen gemeinsam an einem runden Tisch und verbringen ihre Zeit mit Essen und Denken. Jeder hat vor sich einen Teller und zu seiner linken und rechten jeweils eine Gabel (insgesamt fünf). Da die Philosophen zwei Gabeln benötigen um zu essen, können sie nicht gleichzeitig speisen. Hierbei können Verklemmungen auftreten.

Semaphoren dienen dem geregelten Zugriff auf gemeinsame Betriebsmittel. Eine Semaphore verfügt über einen Zähler und atomare Methoden um diesen Zähler zu erhöhen und zu verringern. Ist der Wert des Zählers kleiner Null, wird der zugreifende Prozess in eine Warteschlange eingereiht.

```
import java.util.concurrent.Semaphore;  
// Semaphore mit Maximalwert 2 anlegen  
Semaphore semaphore = new Semaphore(2, true);  
// Semaphore um 1 verringern  
semaphore.acquire();  
// Semaphore um 2 verringern  
semaphore.acquire(2);  
// Semaphore um 1 erhöhen  
semaphore.release();  
// Semaphore um 2 erhöhen  
semaphore.release(2);
```

- a) Implementieren Sie eine Zugriffsstrategie bei der jeder Philosoph jeweils zuerst die rechte, dann die linke Gabel aufnimmt. Das Aufnehmen einer Gabel, das Essen und das Denken benötigen jeweils eine gewisse Zeit. Diese Strategie ist nicht verklemmungsfrei. Wählen sie die benötigten Zeiten so aus, dass eine Verklemmung deutlich wird.
- b) Um die Verklemmung zu vermeiden soll nun ein Philosoph eine Gabel nur dann aufnehmen können, wenn nicht bereits alle anderen Philosophen eine Gabel aufgenommen haben. Implementieren Sie diese Zugriffsstrategie unter Verwendung von Semaphoren und überzeugen Sie sich von der Verklemmungsfreiheit.