

VO 182.711
Prüfung Betriebssysteme

18. Jänner 2013

KNr.

MNr.

Zuname, Vorname

Ges.)(100)

1.)(35)

2.)(20)

3.)(45)

Zusatzblätter:

Bitte verwenden Sie nur dokumentenechtes Schreibmaterial!

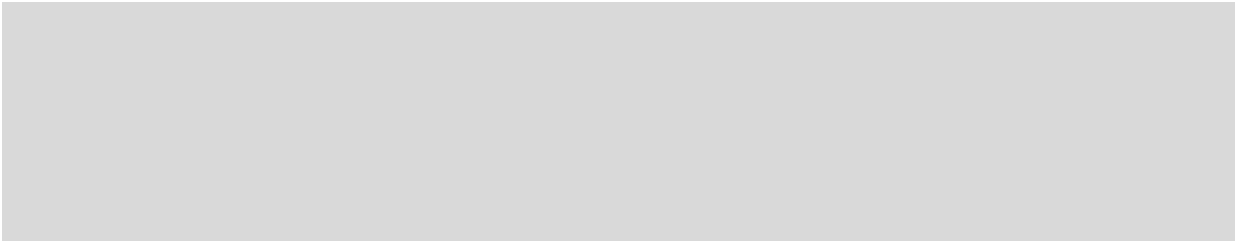
1 Synchronisation mit Semaphoren (35)

Ein Programm soll den Besuch eines Tanzlokals durch Tänzer und Tänzerinnen mit Hilfe von Prozessen simulieren. Die Anzahl der Tanzpaare, die sich gleichzeitig im Lokal befinden dürfen, ist mit 75 beschränkt. Außerdem gibt es einige Regeln, die das Verhalten von Tänzern und Tänzerinnen beschreiben.

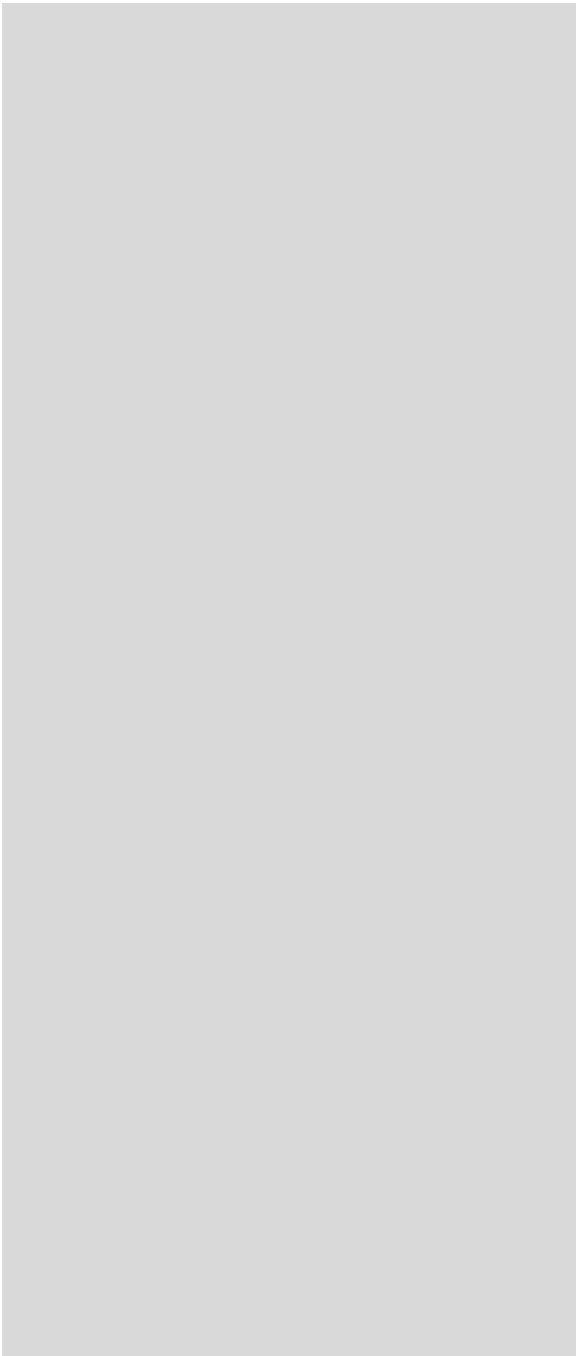
- Die Anreise eines Tänzers bzw. einer Tänzerin wird durch die Funktion *anreise()* realisiert.
- Nach der Anreise treten Tänzer und Tänzerinnen ins Lokal ein, wobei der Eintritt jeweils paarweise (ein Tänzer und eine Tänzerin) erfolgt. Der Tänzer betritt dabei vor der Tänzerin das Lokal. Das Eintreten einer Person wird durch den Aufruf der Funktion *eintreten()* simuliert.
- Nach dem Eintreten ins Lokal tanzen Tänzer und Tänzerin zunächst, was durch den Aufruf der Funktion *tanzen()* beim Tänzer und der Tänzerin simuliert wird.
- Durstig vom Tanz kauft der Tänzer an der Bar Getränke für sich und seine Tanzpartnerin (*kaufeGetränke()*), wobei maximal zwei Tänzer gleichzeitig an der Bar bedient werden.
- Tänzer und Tänzerin trinken dann ihre Getränke (*leereGlas()*). Wenn beide fertig sind, verlassen sie das Lokal (beide rufen die Funktion *bye()* auf).
- Jedesmal, wenn ein Tanzpaar das Lokal verlassen hat, darf ein neues Tanzpaar eintreten.

Schreiben Sie Implementierungen für die Prozesse *Taenzer* und *Taenzerin*, die sich wie beschrieben verhalten. Verwenden Sie ausschließlich Semaphore, um Prozesse zu synchronisieren und die angegebenen Bedingungen und Ereignisfolgen sicherzustellen. Die Verwendung von anderen Synchronisationskonstrukten und globalen Variablen ist nicht erlaubt.

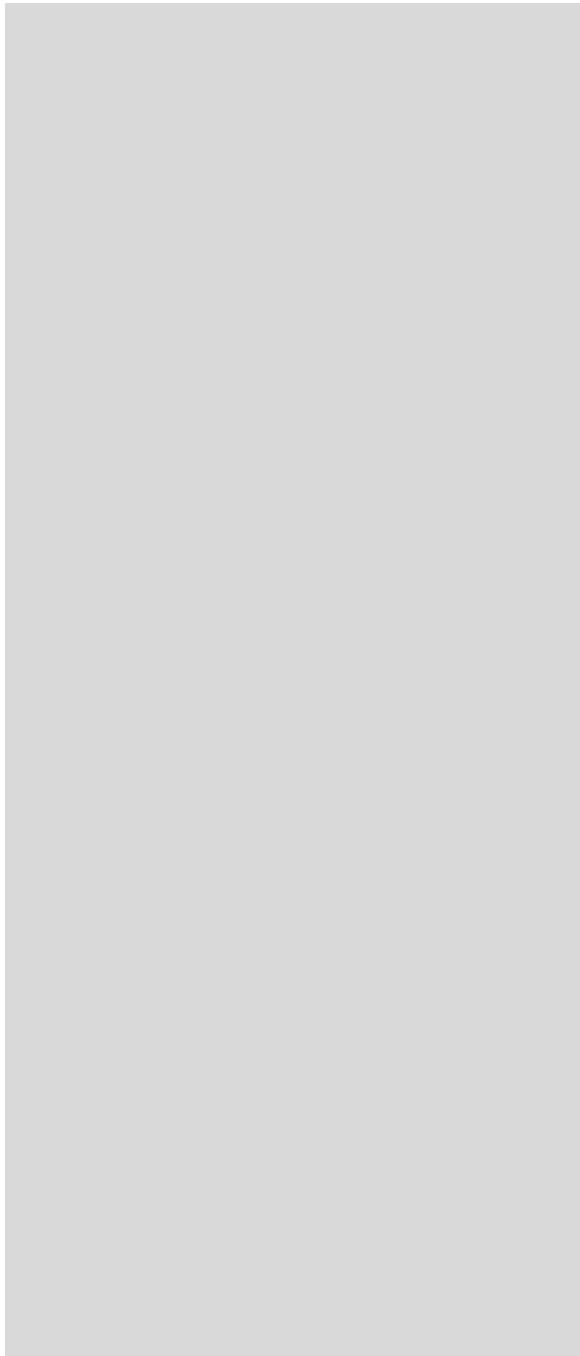
Initialisierungen



`/** Code Taenzer **/`



`/** Code Taenzerin **/`



2 Memory Management (20)

Gegeben ist eine Folge von Speicherreferenzen eines Prozesses. Die Entwicklung des *Working Sets* des Prozesses während der Ausführung soll für unterschiedliche *Window Sizes* (Δ) beobachtet werden. Tragen Sie dazu in der unten stehenden Tabelle für jeden Zeitpunkt das Working Set nach der Referenzierung der angegebenen Speicherseite an.

Referenzierte		Window Size, Δ		
log. Zeit	Seite	2	3	4
1	16			
2	17			
3	18			
4	20			
5	22			
6	20			
7	18			
8	22			
9	20			
10	18			
11	23			
12	23			
13	17			
14	24			
15	23			
16	24			

Nehmen Sie an, dass das Lokalitätsverhalten des im Beispiel verwendeten Prozesses typisch für die Prozesse auf einem Rechner sind. Diskutieren Sie anhand der Beobachtungen aus der obigen Tabelle, welchen Wert für Δ Sie bei der Implementierung der Speicherverwaltung für diesen Rechner verwenden würden.

3 Fragen zu Betriebssystemen (45)

Was versteht man unter einem *Process Image*? Erklären Sie, aus welchen Teilen ein Process Image besteht? (4)

Bei der Deadlock-Vermeidung spricht man von einem *Safe State* bzw. einem *Unsafe State*. Erklären Sie die Bedeutung dieser Begriffe. (4)

Wie funktioniert das CPU-Scheduling nach dem *Highest Response Ration Next* Verfahren? Welche Vorteile bzw. Nachteile hat diese Schedulingstrategie? (4)

Welche Informationen werden in den Einträgen einer *Page Table* gespeichert? Nennen und beschreiben Sie diese kurz. (3)

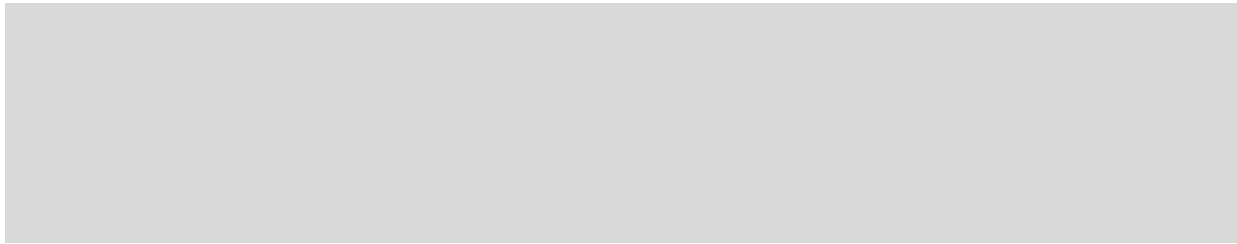
Beschreiben Sie den Aufbau und Verwendung einer *Multilevel Page Table* (eventuell mit Skizze). Warum werden Multilevel Page Tables verwendet? (3)

Was versteht man unter *Synchronous* bzw. *Asynchronous I/O*? Beschreiben Sie die beiden Arten, I/O-Operationen durchzuführen. (3)

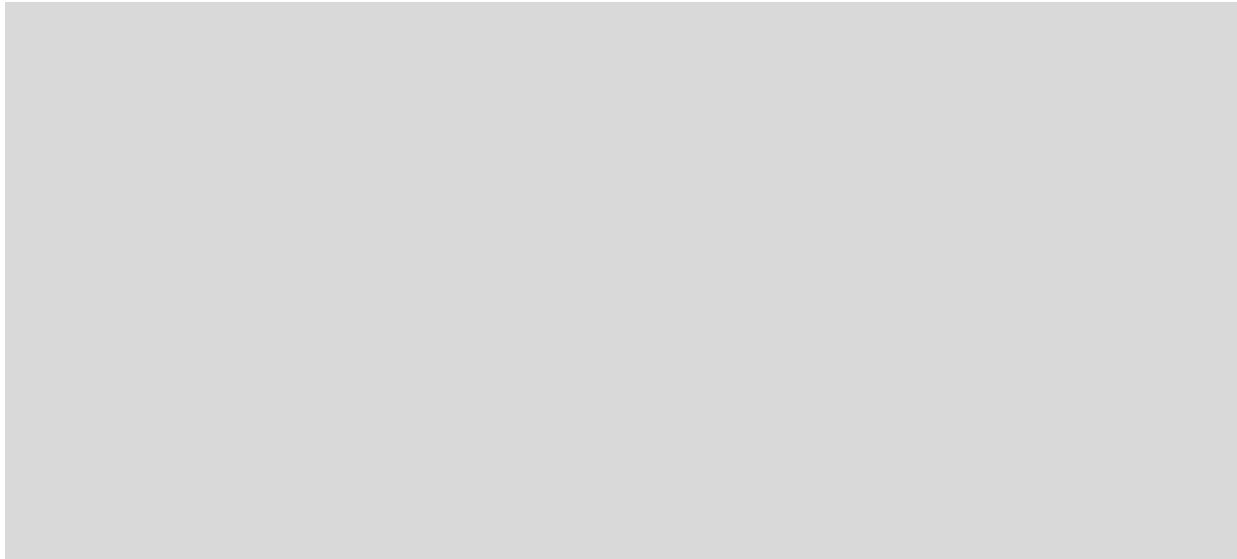
Wie ist ein *i-node* aufgebaut? Welche Informationen enthält er? (4)

Nennen Sie die drei Kategorien von *Security Threats* und beschreiben Sie diese. Geben Sie für jede Kategorie an, welches grundlegende Security-Ziel dadurch bedroht wird. (4)

Wodurch unterscheidet sich ein *Network Operating System* von einem *Distributed Operating System*? (4)



Was versteht man unter dem *Domain Name System (DNS)*? Erklären Sie die Aufgabe von DNS und geben Sie ein Beispiel an, um seine Funktionsweise zu illustrieren. (4)



Beschreiben Sie drei *Naming Strategien*, die in *Verteilten Dateisystemen (DFS)* verwendet werden. (4)

Nennen Sie drei Möglichkeiten, wie *Distributed Mutual Exclusion* realisiert werden kann. Charakterisieren Sie jede dieser Möglichkeiten kurz. (4)

