

## 4 Konzepte von Betriebssystemen (20)

### a) (4)

Geben Sie zwei Abstraktionen an, die ein Betriebssystem zur Verfügung stellt. Beschreiben Sie für jede dieser Abstraktionen, welche Aufgaben sie dem Benutzer abnimmt.

### b) (3)

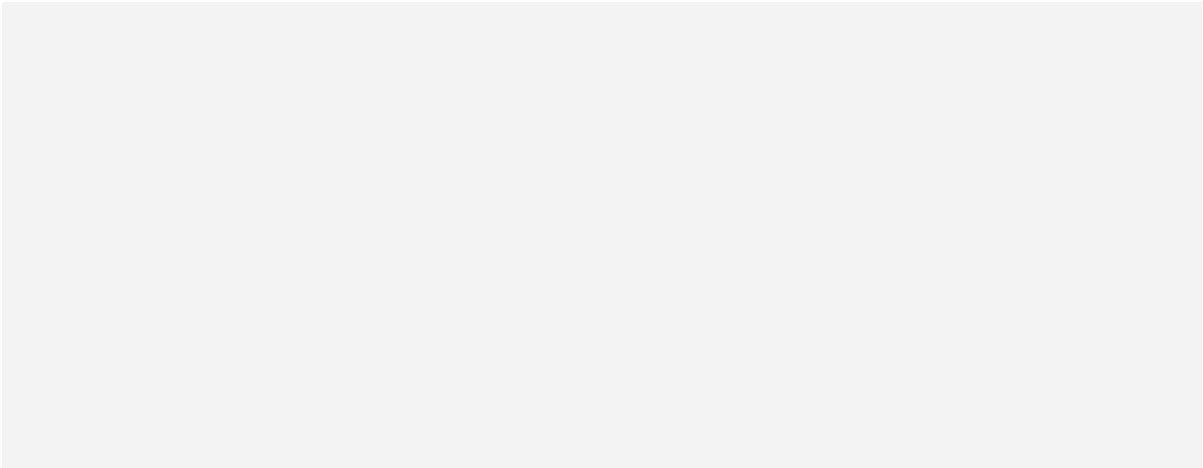
Wodurch gewährleistet ein Betriebssystem die Portabilität von Programmen?

### c) (5)

Geben Sie die drei Fälle an, in denen die Kontrolle über einen Rechner von einem Benutzerprogramm an das Betriebssystem wechselt.

**d) (8)**

Was ist ein Mode Switch? Was ist ein Process Switch? Was passiert bei jedem dieser Switches? Wie stehen Mode Switch und Process Switch in Beziehung?



## 4 Konzepte von Betriebssystemen (20)

### a) (4)

Geben Sie zwei Abstraktionen an, die ein Betriebssystem zur Verfügung stellt. Beschreiben Sie für jede dieser Abstraktionen, welche Aufgaben sie dem Benutzer abnimmt.

### b) (3)

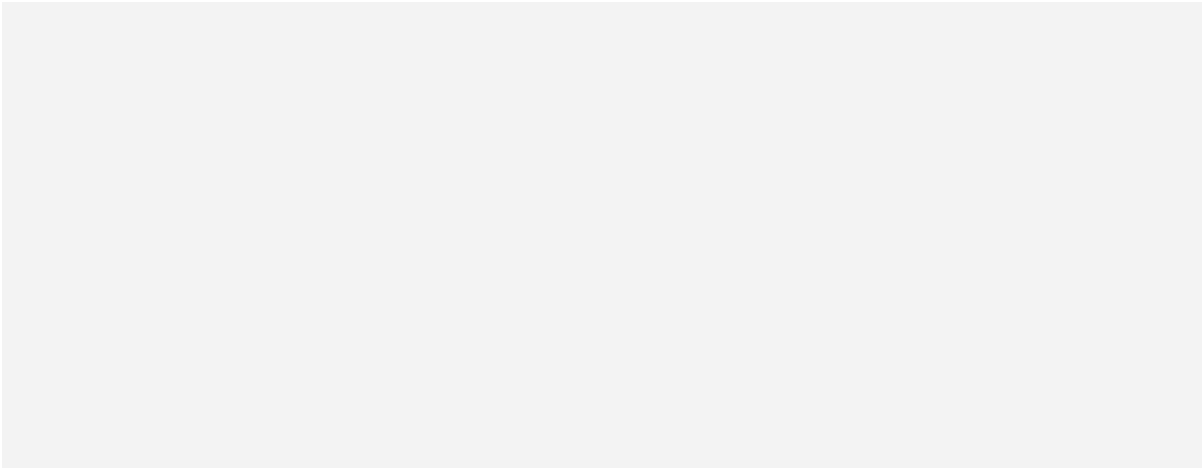
Wodurch gewährleistet ein Betriebssystem die Portabilität von Programmen?

### c) (5)

Geben Sie die drei Fälle an, in denen die Kontrolle über einen Rechner von einem Benutzerprogramm an das Betriebssystem wechselt.

**d) (8)**

Was ist ein Mode Switch? Was ist ein Process Switch? Was passiert bei jedem dieser Switches? Wie stehen Mode Switch und Process Switch in Beziehung?



## 4 BS-Abstraktionen und Prozesse (30)

Nennen Sie drei Abstraktionen, die ein Betriebssystem zur Verfügung stellt. Geben Sie für jede dieser Abstraktionen einerseits die Benutzerwahrnehmung andererseits die zur Realisierung der Abstraktion notwendigen Betriebssystemmechanismen an.

Welche Schritte muss ein Betriebssystem bei einem Process Switch (Context Switch) durchführen?

Nennen Sie die prozessspezifischen Informationen, die ein Betriebssystem im Process Control Block verwaltet.

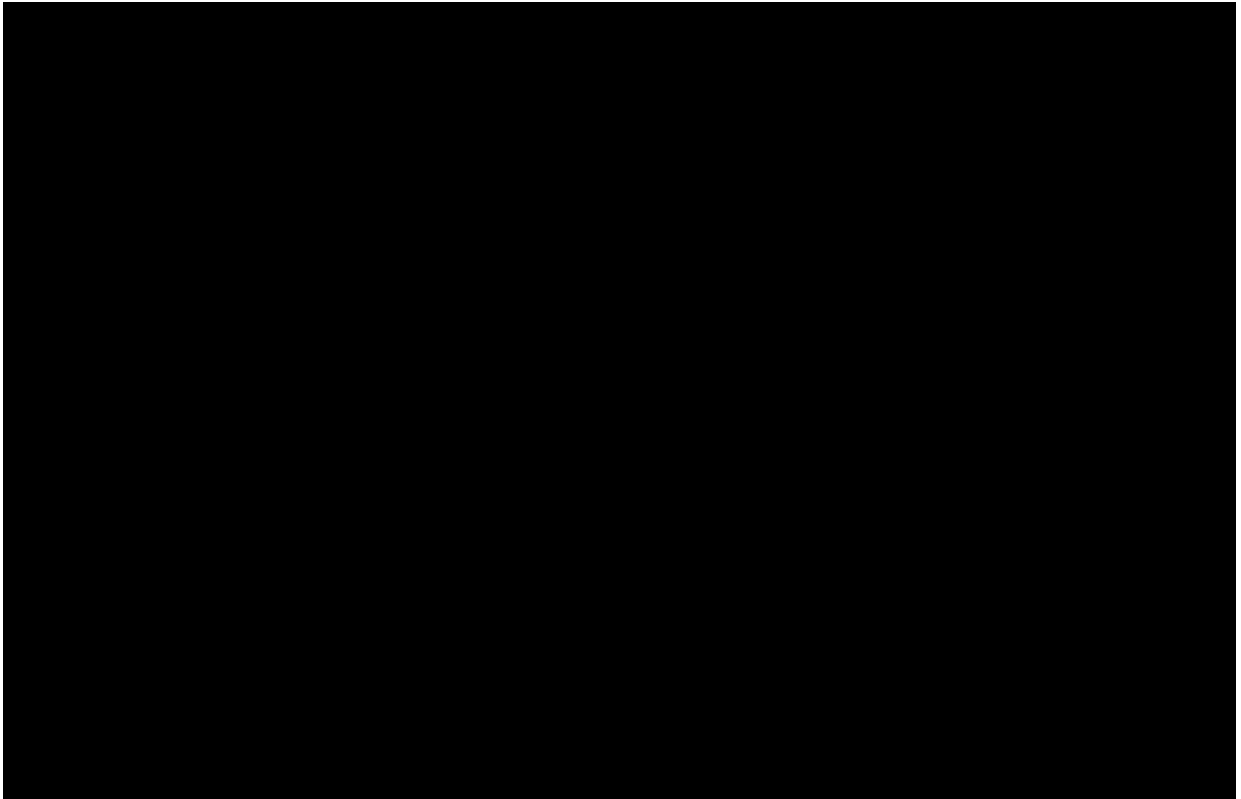
Erklären Sie den Unterschied zwischen den Begriffen *Prozess* und *Thread*. In welcher Beziehung stehen diese beiden Konzepte?

In der Vorlesung wurden zwei Möglichkeiten der Implementierung von Threads genannt. Um welche Implementierungen handelt es sich dabei? Wie machen sich die Unterschiede in der Implementierung für Applikationen bemerkbar?

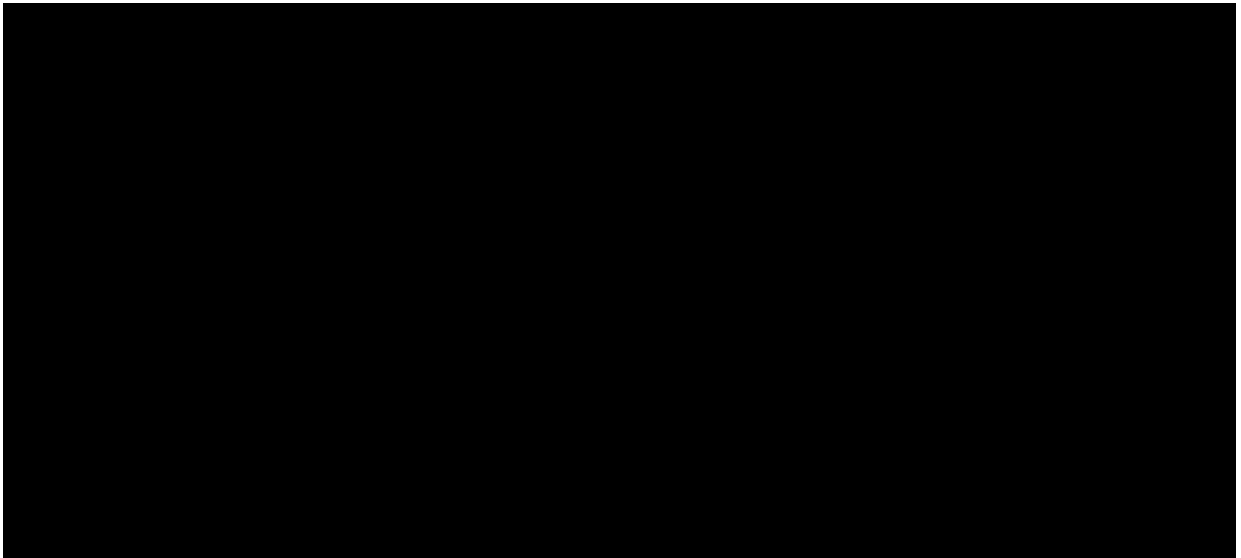
Erklären Sie, was man unter (a) einem *monolithischen Kernel* und (b) einem *Microkernel* versteht.

## 4 Betriebssysteme, Prozesse und Threads (23)

Was versteht man unter einem Microkernel-Betriebssystem? Welche Services stellt ein Microkernel zur Verfügung? Welche Eigenschaften zeichnen ein solches Betriebssystem aus?



Was ist ein *Mode Switch*? Was ist ein *Process Switch*? Erklären Sie die beiden Begriffe. Geben Sie an, wozu diese Switches benötigt werden und erläutern Sie, welcher Zusammenhang zwischen Mode Switch und Process Switch besteht.



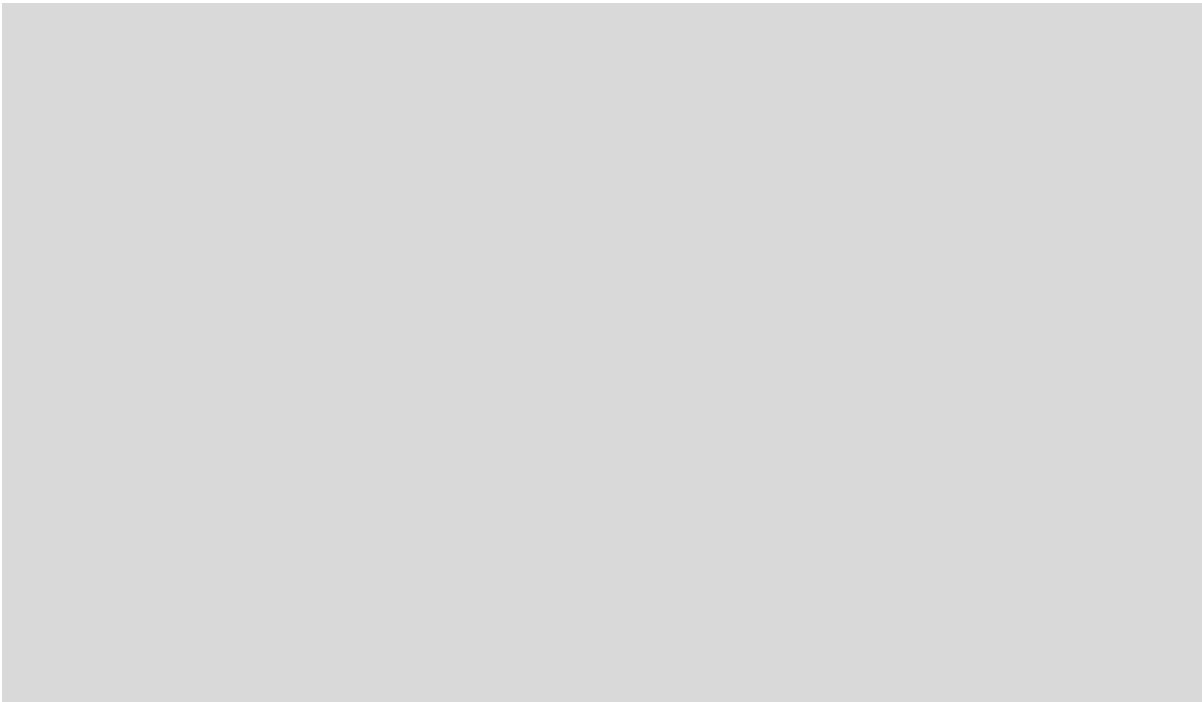
Nennen Sie die drei Kategorien von Ereignissen, mit deren Hilfe das Betriebssystem die Kontrolle über das Computersystem übernimmt. Geben Sie für jede der Kategorien ein Beispiel an.

Beschreiben Sie *User-Level Threads* und *Kernel-Level Threads*. Wodurch unterscheiden sich diese beiden Arten der Thread-Implementierung?

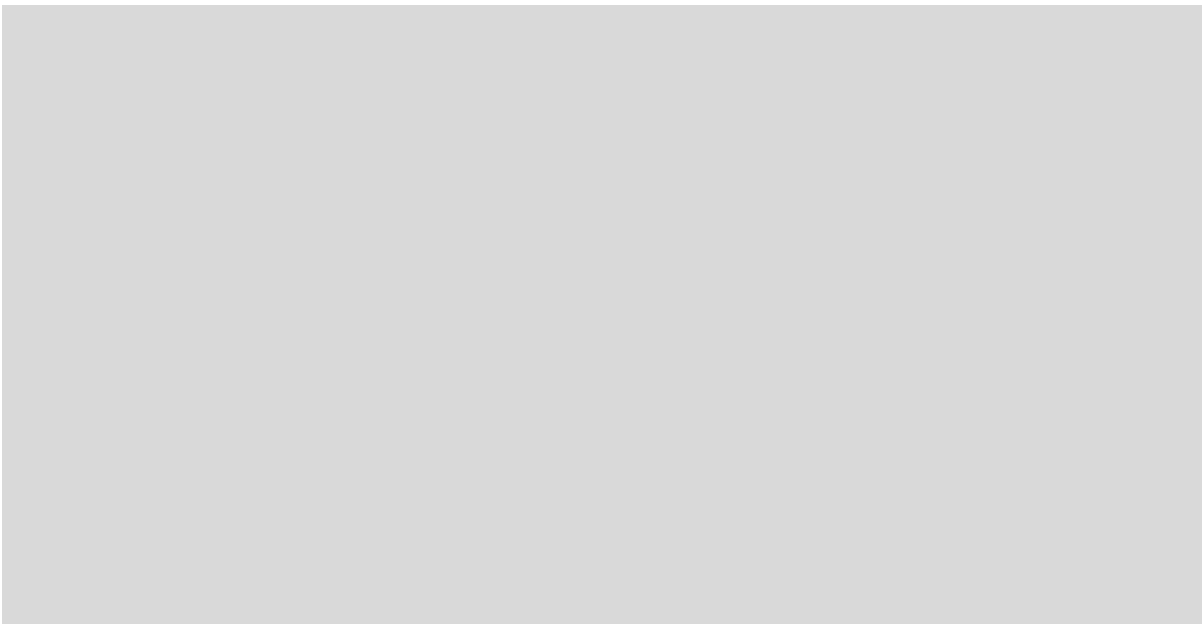


### 3 Fragen zu Betriebssystemen (50)

Zeichnen Sie das Zustandsdiagramm eines Prozesses, beginnend mit der Entstehung bis zur Beendigung des Prozesses. Geben Sie die Namen der Zustände an und beschreiben Sie kurz jeden Zustand und Zustandsübergang. (5)



Welche Strategien zur Vorbeugung gegen Deadlocks bzw. zur Vermeidung von Deadlocks gibt es? Beschreiben Sie diese kurz. (5)



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

Erklären Sie den Begriff des *Multithreading*. Geben Sie Probleme bei der Verwendung von Threads an. (3)

Nennen Sie die Arten von Optimierungszielen, die ein Scheduler beim Prozess-Scheduling verfolgen kann und geben Sie jeweils zwei Beispiele an. (4)

Was versteht man unter einem *Monitor* zur Prozesssynchronisation? Nennen Sie die wichtigsten Komponenten und Eigenschaften eines Monitors. (4)

Was versteht man unter dem Begriff *Relocation*? (2)

Beim Paging ist ein optimaler Seitenersetzungsalgorithmus bekannt. Beschreiben Sie diesen und geben Sie an, warum er in der Praxis nicht verwendet wird. (3)

Welche Möglichkeiten kennen Sie, mit denen in Paging-Systemen Speicherschutz realisiert werden kann? (4)

Was versteht man unter der Working-Set Strategie? Beschreiben Sie deren Funktionsweise und wie diese Strategie zur Optimierung eines Paging-Systems eingesetzt werden kann. (5)

Nennen Sie Möglichkeiten, wie die Blockbelegung von Dateien auf einer Festplatte repräsentiert werden kann. Geben Sie Vor- und Nachteile der angegebenen Organisationsformen an. (4)

Erklären Sie die Begriffe *absoluter Pfadname* und *relativer Pfadname* und geben Sie jeweils ein Beispiel an. (4)

Die Implementierung einer Zugriffsmatrix kann in der Form von *Access Control Lists* oder *Capability Lists* erfolgen. Erklären Sie diese Begriffe. (4)

### 3 Fragen zu Betriebssystemen (45)

Erklären Sie die drei Begriffe *Deadlock*, *Livelock* und *Starvation*. (5)

Handwritten answer:

*Deadlock*: Zustand, in dem zwei oder mehr Prozesse auf Ressourcen warten, die von anderen Prozessen gehalten werden, die wiederum auf Ressourcen warten, die von den ersten Prozessen gehalten werden. Es gibt keine Möglichkeit, den Zustand zu beenden.

*Livelock*: Zustand, in dem zwei oder mehr Prozesse auf Ressourcen warten, die von anderen Prozessen gehalten werden, die wiederum auf Ressourcen warten, die von den ersten Prozessen gehalten werden. Es gibt keine Möglichkeit, den Zustand zu beenden.

*Starvation*: Zustand, in dem ein Prozess für eine unendliche Zeit auf eine Ressource warten muss, weil andere Prozesse die Ressource monopolisieren.

Worin liegt der grundlegende Unterschied zwischen *Prozessen* und *Threads*? Welcher Vorteil ergibt sich aus der Einführung von Threads für den Benutzer und worauf muss der Benutzer achten? (4)

Handwritten answer:

*Prozessen* und *Threads* unterscheiden sich in der Weise, wie sie Ressourcen des Betriebssystems nutzen. Ein *Prozess* ist eine unabhängige Einheit der Ausführung, die einen eigenen Adressraum und Systemressourcen benötigt. Ein *Thread* ist eine Teilinstanz eines Prozesses, die denselben Adressraum und Systemressourcen teilt. Der Vorteil von Threads liegt in der besseren Auslastung der CPU und der Möglichkeit, mehrere Aufgaben gleichzeitig auszuführen. Der Benutzer muss jedoch auf die Thread-Sicherheit achten, um Datenkorruption zu vermeiden.

Was versteht man unter *Virtual Memory Management*? Welche Vorteile bietet es? (5)

Handwritten answer:

*Virtual Memory Management* ist die Verwaltung des virtuellen Hauptspeichers. Es ermöglicht die Ausführung von Programmen, die mehr Hauptspeicher benötigen, als physisch vorhanden ist, indem Teile des Hauptspeichers auf der Festplatte (Swap) gespeichert werden. Die Vorteile sind: Erhöhung der Flexibilität bei der Speichernutzung, Vermeidung von Konflikten zwischen Programmen und Erhöhung der Stabilität des Systems.

Beschreiben Sie Aufgabe und Funktion eines *Translation Lookaside Buffers*? Worauf hat man bei der Betriebssystemimplementierung bei einem Process Switch zu achten, wenn man einen Translation Lookaside Buffer verwendet? (4)

Ein TLB (Translation Lookaside Buffer) ist eine kleine, schnelle Cache-Struktur, die die aktuellen Übersetzungen von virtuellen Adressen in physische Adressen speichert. Seine Aufgabe ist es, die Übersetzung von virtuellen Adressen in physische Adressen zu beschleunigen, indem es die Übersetzungstabellen des Hauptspeichers umgeht. Wenn ein Prozess eine virtuelle Adresse benötigt, sucht der TLB nach dieser Adresse. Wenn er sie findet, kann er die physische Adresse sofort zurückgeben. Wenn er sie nicht findet, muss der Prozessor die Übersetzungstabelle des Hauptspeichers durchsuchen, um die physische Adresse zu finden. Nach der Übersetzung wird das Ergebnis im TLB gespeichert, um zukünftige Übersetzungen zu beschleunigen. Bei einem Prozess-Switch muss der TLB geleert werden, um die Übersetzungen des neuen Prozesses zu speichern. Wenn der TLB nicht geleert wird, kann der alte Prozess die physische Adresse des neuen Prozesses erhalten, was zu einer Speicher-Kollision führen kann.

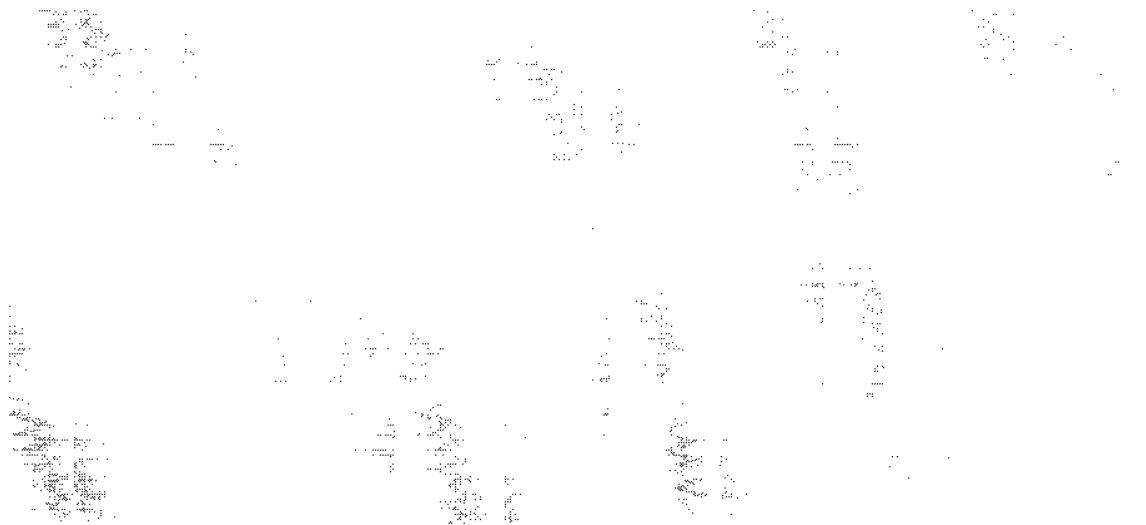
Was versteht man unter *Thrashing*? (2)

Thrashing ist ein Zustand, in dem ein System so viel Zeit damit verbringt, Daten zwischen dem Hauptspeicher und dem Festplatte zu verschieben, dass es keine Zeit mehr hat, um die eigentlichen Aufgaben des Benutzers zu erledigen. Dies tritt auf, wenn die Arbeitslast des Systems größer ist als die Kapazität des Hauptspeichers. In diesem Zustand wird die Leistung des Systems stark beeinträchtigt, da die Zeit, die für die Datenverschiebung benötigt wird, die Zeit, die für die Ausführung der Aufgaben benötigt wird, übersteigt. Thrashing kann durch eine Erhöhung der Hauptspeicherkapazität oder durch eine Reduzierung der Arbeitslast des Systems verhindert werden.

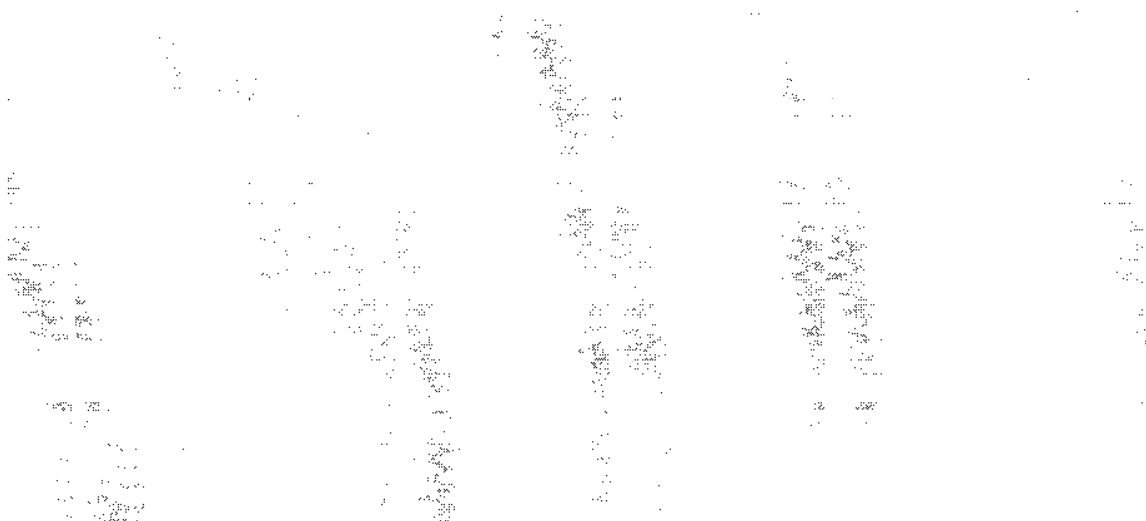
Wie funktioniert *Round Robin* Scheduling? Welchen wichtigen Parameter gibt es bei diesem Verfahren? Wie wird man diesen Parameter günstiger Weise wählen? (4)

Round Robin Scheduling ist ein Verfahren, bei dem die CPU-Zeit zwischen den Prozessen in Runden aufgeteilt wird. Jeder Prozess erhält eine bestimmte Zeit, um seine Aufgabe auszuführen, bevor er in die Warteschlange zurückgeführt wird. Die Zeit, die einem Prozess für eine Runde zugewiesen wird, wird als Zeitquantum bezeichnet. Die Wahl des Zeitquantums ist ein wichtiger Parameter, der die Leistung des Scheduling-Verfahrens beeinflusst. Ein zu kleines Zeitquantum führt zu einer hohen Anzahl von Context Switches, was die Leistung des Systems beeinträchtigt. Ein zu großes Zeitquantum führt zu einer langen Wartezeit für die Prozesse, die in der Warteschlange stehen. Ein günstiger Wert für das Zeitquantum kann durch Experimente oder durch die Analyse der Last des Systems bestimmt werden.

Was versteht man unter *Buffering*? Welche Vorteile bietet es, wo liegen seine Grenzen und worauf hat man bei der Verwendung von Puffern bei der Betriebssystemimplementierung zu achten? (5)



Wie ist der Inhalt einer (mechanischen) Festplatte typischer Weise organisiert? (4)



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





Was beschreibt das Modell von Bell und LaPadula? Geben Sie die vom Modell geforderten Eigenschaften an. (4)

Das Modell von Bell und LaPadula ist ein Sicherheitsmodell für die Vertraulichkeit von Informationen. Es beschreibt die Eigenschaften, die ein System erfüllen muss, um die Vertraulichkeit zu gewährleisten. Die Eigenschaften sind:

- Simple Security Property:** Ein Subjekt darf nur auf ein Objekt zugreifen, wenn seine Vertraulichkeitsstufe nicht höher ist als die des Objekts.
- \*Property (Star Property):** Ein Subjekt darf nur auf ein Objekt zugreifen, wenn seine Vertraulichkeitsstufe nicht höher ist als die des Objekts.
- Transitive Property:** Wenn ein Subjekt auf ein Objekt zugreifen darf, dann darf es auch auf alle Objekte zugreifen, die von diesem Objekt abgeleitet sind.

Nennen Sie Design Prinzipien für die Konstruktion von sicheren Systemen. Geben Sie für jede Regel ein Beispiel an. (4)

Die Design Prinzipien für die Konstruktion von sicheren Systemen sind:

- Prinzip der Vertraulichkeit:** Ein System muss sicherstellen, dass Informationen nicht an Unbefugte weitergegeben werden.
- Prinzip der Integrität:** Ein System muss sicherstellen, dass Informationen nicht verändert werden können.
- Prinzip der Verfügbarkeit:** Ein System muss sicherstellen, dass Informationen jederzeit verfügbar sind.
- Prinzip der Authentizität:** Ein System muss sicherstellen, dass Informationen von der richtigen Quelle stammen.