

Nachname/*Last name*

Vorname/*First name*

Matrikelnr./*Matriculation no*

Hauptklausur

10.03.2017

- Bitte tragen Sie zuerst auf dem Deckblatt Ihren Namen, Ihren Vornamen und Ihre Matrikelnummer ein. Tragen Sie dann auf den anderen Blättern (auch auf Konzeptblättern) Ihre Matrikelnummer ein.
Please fill in your last name, your first name, and your matriculation number on this page and fill in your matriculation number on all other pages (including draft pages).
- Die Prüfung besteht aus 15 Blättern: Einem Deckblatt und 14 Aufgabenblättern mit insgesamt 5 Aufgaben.
The examination consists of 15 pages: One cover sheet and 14 sheets containing 5 assignments.
- Es sind keinerlei Hilfsmittel erlaubt!
No additional material is allowed.
- Die Prüfung gilt als nicht bestanden, wenn Sie versuchen, aktiv oder passiv zu betrügen.
You fail the examination if you try to cheat actively or passively.
- Wenn Sie zusätzliches Konzeptpapier benötigen, verständigen Sie bitte die Klausuraufsicht.
If you need additional draft paper, please notify one of the supervisors.
- Bitte machen Sie eindeutig klar, was Ihre endgültige Lösung zu den jeweiligen Teilaufgaben ist. Teilaufgaben mit widersprüchlichen Lösungen werden mit 0 Punkten bewertet.
Make sure to clearly mark your final solution to each question. Questions with multiple, contradicting answers are void (0 points).

Die folgende Tabelle wird von uns ausgefüllt! *The following table is completed by us!*

Aufgabe	1	2	3	4	5	Total
Max. Punkte	12	12	12	12	12	60
Erreichte Punkte						
Note						

Aufgabe 1: Grundlagen

Assignment 1: Basics

- a) Skizzieren Sie die wichtigsten Schritte bei der Behandlung eines Interrupts durch die CPU und das Betriebssystem.

2 pt

Outline the most important steps during interrupt handling by the CPU and the operating system.

- b) Nennen Sie zwei privilegierte CPU-Operationen auf x86 und erklären Sie jeweils, warum diese privilegiert sein müssen. Eine Angabe der Assemblermnemonics ist nicht nötig. Inverse Operationen zählen nicht.

2 pt

Give two privileged CPU operations on x86 and explain why each of them needs to be privileged. You do not need to provide the assembler mnemonics. Inverse operations do not count.

- c) Treten CPU-Ausnahmen synchron zum Programmfluss auf?

0.5 pt

Do CPU exceptions occur synchronously with the program flow?

Ja / Yes

Nein / No

- d) Betrachten Sie ein Betriebssystem, in dem Prozesse und der Kernel jeweils dedizierte, vollständige Adressräume besitzen. Bei Ausführung der Trap-Instruktion wird automatisch in den Adressraum des Kernels sowie auf den jeweiligen Kernelstack gewechselt.

Consider an operating system in which processes and the kernel each possess a dedicated, full address space. When executing the trap instruction, the system automatically switches to the kernel's address space and to the respective kernel stack.

Ein Systemaufruf erfordert mehr Argumente, als es CPU-Register gibt. Erläutern Sie, wie die Parameterübergabe in diesem System funktionieren kann.

1.5 pt

A system call requires more arguments than there are CPU registers available. Explain how the parameters could be passed in this system.

Nennen Sie einen Vor- und einen Nachteil von einem dedizierten Kerneladressraum auf 32 Bit Systemen. Wie bewerten Sie dieses Design auf 64 Bit Systemen?

1.5 pt

Give an advantage and a disadvantage of a dedicated kernel address space on 32 bit systems. How do you rate this design on 64 bit systems?

Aufgabe 2: Prozesse und Threads

Assignment 2: Processes and Threads

- a) Erklären Sie den Unterschied zwischen einem Programm und einem Prozess.

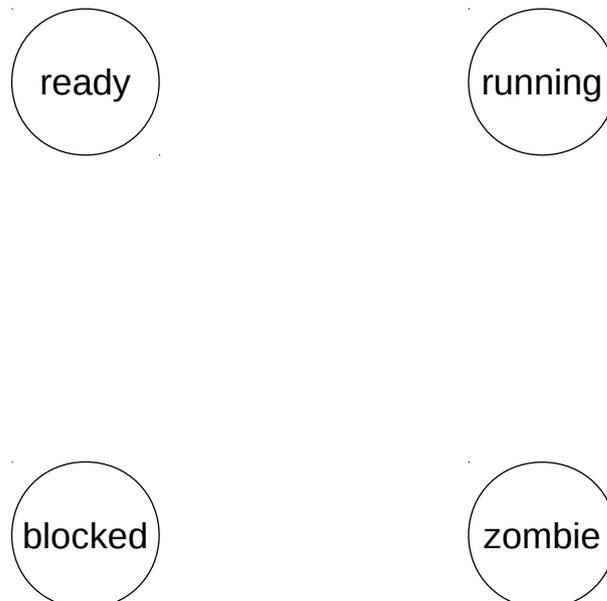
1 pt

Explain the difference between a program and a process.

- b) Nehmen Sie an, dass ein Betriebssystem die vier Prozesszustände "bereit" ("ready"), "rechnend" ("running"), "blockiert" ("blocked") und "zombie" unterstützt. Stellen Sie grafisch dar, zwischen welchen Zuständen Übergänge möglich sind und beschriften Sie jede Kante mit einem Ereignis, das den jeweiligen Übergang auslöst.

2.5 pt

Consider an operating system, which supports the four process states "ready", "running", "blocked", and "zombie". Depict the possible state transitions and label each edge with an event causing the transition.



- c) Erläutern Sie die Funktion des *PCB* und des *TCB*. Geben Sie dabei je zwei Einträge an, die in der jeweiligen Struktur gespeichert werden.

2 pt

Explain the purpose of the PCB and the TCB. For each structure, give two pieces of information that are stored in it.

- d) Was versteht man unter einem *User-Level Thread (ULT)*? Welche aus der Vorlesung bekannten Threadmodelle verwenden ULTs?

1.5 pt

What is a User-Level Thread (ULT)? Which thread models presented in the lecture employ ULTs?

- e) Ein Webserver verwendet einen statischen Pool von Prozessen, um Clientverbindungen parallel zu verarbeiten. Nennen Sie einen Vor- und einen Nachteil dieses Ansatzes gegenüber der Verwendung einer gleichen Anzahl von Threads.

2 pt

A webserver uses a fixed pool of processes to handle connections from multiple clients in parallel. Give an advantage and a disadvantage of this scheme compared to using the same number of threads.

- f) Linux enthält neben dem präemptiven MLFQ-basierten Scheduler einen nicht-präemptiven FCFS-Scheduler mit der Eigenschaft, dass FCFS-Threads immer Vorrang vor MLFQ-Threads haben. Prozesse mit Administratorrechten können entscheiden, unter welchem Scheduler ihre Threads laufen. Prozesse ohne Administratorrechte verwenden immer den MLFQ-Scheduler.

Besides the preemptive MLFQ-based scheduler, the Linux kernel also contains a non-preemptive FCFS scheduler. Threads scheduled by the FCFS scheduler always have priority over threads under the MLFQ scheduler. Processes with administrator privileges can choose which scheduler their threads use. Processes without administrator privileges always use the MLFQ scheduler.

Für welche Art von Anwendungen ist es sinnvoll, den FCFS-Scheduler zu verwenden? Begründen Sie Ihre Antwort.

2 pt

For which type of application is the FCFS scheduler beneficial? Justify your answer.

Warum darf der FCFS-Scheduler nur von Anwendungen mit Administratorrechten verwendet werden?

1 pt

Why are only applications with administrator privileges allowed to use the FCFS scheduler?

**Total:
12.0pt**

Aufgabe 3: Koordination und Kommunikation von Prozessen
Assignment 3: Process Coordination and Communication

- a) Erläutern Sie kurz den Begriff *kritischer Abschnitt*. Warum müssen diese geschützt werden?

2 pt

Briefly explain the term critical section. Why do they need to be protected?

- b) Gegeben sei ein Einprozessorsystem, das alle Threads präemptiv im Kernelmodus ausführt. Welche der notwendigen Bedingungen für eine gültige Lösung des Problems kritischer Abschnitte werden erfüllt, wenn zum Schutz von kritischen Abschnitten Interrupts deaktiviert werden? Welche nicht? Erläutern Sie jeweils warum.

4.5 pt

Consider a uni-processor system in which all threads are executing with preemption in kernel mode. Which of the requirements for a valid solution of the critical section problem does deactivating interrupts fulfill, which not? For each requirement explain why / why not.

- c) Betrachten Sie die folgende Implementierung eines Spinlocks. Funktioniert diese wie gewünscht? Falls ja, begründen Sie, warum. Falls nicht, geben Sie einen Ablauf an, der zu unerwünschtem Verhalten führt.

2.5 pt

Consider the following implementation of a spinlock. Does it work as intended? If so, explain why. If not, describe a flow of execution that leads to undesired behavior.

```

1  bool locked = false           10 function release () {
2                                     11     locked = false
3  function acquire () {          12 }
4     retry:
5     if (locked == true)
6         goto retry
7
8     locked = true
9 }
```

- d) Unter welcher Voraussetzung ist die Verwendung von Spinlocks zur Synchronisation zweier Threads sinnvoll?

1 pt

Under which condition does the use of spinlocks to synchronize two threads makes sense?

- e) Gegeben sei ein System, das blockierende (d. h. synchrone) Interprozesskommunikation (IPC) unterstützt. Wie kann auf einem solchen System ein blockierendes Lock für beliebig viele parallele Threads mit IPC realisiert werden?

2 pt

Consider a system that supports blocking (i.e., synchronous) inter-process communication (IPC). How can you realize a blocking lock with IPC for an arbitrary number of parallel threads?

**Total:
12.0pt**

Aufgabe 4: Speicher

Assignment 4: Memory

- a) Nennen Sie zwei Situationen, in denen das Betriebssystem einen Prozess bei einem Seitenfehler in der Regel beendet.

1 pt

Give two situations in which the operating system usually terminates a process on a page fault.

- b) Erklären Sie, warum sich die LRU-Seitenersetzungsstrategie auf x86 nicht effizient umsetzen lässt. Durch welche Strategie kann LRU approximiert werden? Begründen Sie Ihre Antwort.

3 pt

Explain why LRU page replacement cannot be efficiently done on x86. Which strategy can be used to approximate LRU? Explain your answer.

d) Erläutern Sie die Begriffe *interne* und *externe Fragmentierung*.

2 pt

Explain the terms internal and external fragmentation.

e) Erläutern Sie, wie sich die Seitengröße auf die Trefferquote des TLB auswirkt.

1 pt

Explain how the page size affects the hit rate of the TLB.

f) Beschreiben Sie, warum es beim Demand-Paging zu einer erhöhten Anzahl von Seitenfehlern kommen kann.

1 pt

Describe why demand paging may lead to an increased number of page faults.

**Total:
12.0pt**

Aufgabe 5: I/O, Hintergrundspeicher und Dateisysteme

Assignment 5: I/O, Secondary Storage, and File Systems

- a) Nennen Sie die drei in der Vorlesung genannten Orte im Dateisystem, an denen der Typ von Dateien kodiert werden kann.

1 pt

List the three places in the file system, where, according to the lecture, it is possible to encode file types.

- b) Erklären Sie den Unterschied zwischen Mandatory und Advisory File Locks.

1 pt

Explain the difference between mandatory and advisory file locks.

- c) Während Sektoren oft nur 512 Bytes umfassen, verwalten viele Dateisysteme den Speicherplatz in wesentlich größeren Blöcken (z. B. 4 KiB bei FAT32). Nennen Sie einen Vor- und einen Nachteil von größeren Blöcken fester Größe gegenüber kleineren Blöcken.

1 pt

Sectors are usually 512 bytes large, whereas many file systems use blocks of significantly larger size (e.g., 4 KiB in FAT32). Give an advantage and a disadvantage of such larger blocks of fixed size compared to smaller blocks.

- e) Nennen Sie eine Situation, in der es notwendig ist, dass bei einer Schreiboperation in eine Datei zuvor existierende Dateiinhalte der selben Datei vom Blockgerät gelesen werden müssen. Begründen Sie Ihre Antwort.

2 pt

Describe a situation, where a write operation into a file requires the OS to first read existing file contents of the same file from the block device. Justify your answer.

- f) Warum kann es dazu kommen, dass eine Leseoperation keine Zugriffe auf das darunter liegende Blockgerät verursacht?

1 pt

Why do some file read operations trigger no accesses to the underlying block device?

- g) Muss in einem RAID 3 beim Schreiben eines einzelnen Blocks zwangsweise auf alle Festplatten des Verbunds zugegriffen werden? Begründen Sie Ihre Antwort.

1 pt

When writing a single block in a RAID 3, do all disks in the array need to be accessed? Justify your answer.

**Total:
12.0pt**