

# Musterlösung

18. 03. 2014

Alle Punkteangaben ohne Gewähr!

- Bitte tragen Sie zuerst auf dem Deckblatt Ihren Namen, Ihren Vornamen und Ihre Matrikelnummer ein. Tragen Sie dann auf den anderen Blättern (auch auf dem Konzeptblatt) 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 the draft page).*
- Die Prüfung besteht aus 12 Blättern: Einem Deckblatt und 11 Aufgabenblättern mit insgesamt 5 Aufgaben.  
*The examination consists of 12 pages: One cover sheet and 11 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).*
- Wir werden Punkte abziehen, falls korrekte Antworten auch inkorrekte oder irrelevante Informationen enthalten. Bitte schreiben Sie nicht einfach möglichst viel hin, in der Hoffnung, das richtige Schlagwort zu treffen.  
*We will take off points if a correct answer also includes incorrect or irrelevant information. Do not write down everything you know in hopes of saying the correct buzz word.*

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

Aufgabe	1	2	3	4	5	Total
Max. Punkte	12	13	12	12	11	60
Erreichte Punkte						
Note						

**Aufgabe 1: Zum Aufwärmen / Assignment 1: Warmup**

- a) Wie werden die folgenden Operationen in der Programmiersprache C geschrieben? Nehmen sie an, dass die folgenden Definitionen gelten: `int a, b, A[42];`. Drücken Sie die letzte Antwort ohne eckige Klammern aus!

**2 pt**

*How are the following operations expressed in the C programming language? Assume the following definitions: `int a, b, A[42];`. Express the last answer without square braces!*

**Lösung:**

<i>Bitwise exclusive or of a and b</i>	$a \wedge b$
<i>Bitwise not of a</i>	$\sim a$
<i>Address of a</i>	$\&a$
<i>Address of A[23]</i>	$A+23$

- b) Beachten Sie Listing 1. In welchem Adressraumsegment werden die folgenden Variablen jeweils angelegt?

**4 pt**

*Consider Listing 1. In which address space segment will the following variables be allocated respectively?*

Listing 1

```

1 int a = 23;
2 int f ()
3 {
4     int * p = malloc( 42 * sizeof(int) );
5 }
```

**Lösung:**

<i>a</i>	<i>data segment</i>
<i>f</i>	<i>text segment</i>
<i>p (pointer)</i>	<i>stack segment</i>
<i>*p (content that p points to)</i>	<i>heap segment</i>

- c) Unterstreichen Sie diejenigen Funktionen, welche man als Systemaufruf bezeichnen würde!

**1 pt**

*Underline those functions that one would consider system calls!*

**Lösung:**

`getopt`   `read`   `lseek`   `malloc`   `memset`   `mmap`

- d) Ordnen Sie die folgenden Dinge aufsteigend ihrer Dauer nach an:  
(a) Kontextwechsel, (b) Funktionsaufruf, (c) Speicherzugriff (1 Byte), (d) SSD Leseoperation

**2 pt**

*Order the following entities by the time they take in increasing order:  
(a) Context switch, (b) Function call, (c) Memory access (1 Byte), (d) SSD read*

**Lösung:**

$c \Rightarrow b \Rightarrow a \Rightarrow d$

- e) Welche der folgenden Aussagen sind korrekt, welche sind inkorrekt?  
(falsches Kreuz: -1P, kein Kreuz: 0P, korrektes Kreuz: 1P)

**3 pt**

*Which of the following statements are correct, which are incorrect?  
(incorrectly marked: -1P, not marked: 0P, correctly marked: 1P)*

korrekt/ <i>correct</i>	inkorrekt/ <i>incorrect</i>
----------------------------	--------------------------------



$(a \& 1) == (a \% 2)$

*Assume, that a is an unsigned integer.*



Die Trap-Instruktion ist privilegiert.

*The trap instruction is privileged.*



In Linux: Wenn die Datei f in Prozess p geöffnet ist und ein anderer Prozess f löscht, dann löst eine folgende Lese- oder Schreibeoperation von p eine Ausnahme aus.

*In Linux: When file f is open in process p and another process deletes f then a subsequent read and write operation in p raises an exception.*

**Total:  
12.0pt**

## Aufgabe 2: Prozesse und Threads / Assignment 2: Processes and Threads

- a) Beschreiben Sie kurz den Unterschied zwischen einem Prozess und einem Thread im Kontext der Vorlesung.

1 pt

*Briefly describe the difference between a process and a thread in the context of the lecture.*

### Lösung:

*A process consists of an address space (0.5 P) and all threads executing inside that address space. (0.5 P) A thread is thus a component of a process.*

*(Another potential answer: A process is always a scheduling entity of the OS. (0.5 P) The thread model determines whether threads are OS entities. (0.5 P) )*

- b) Gegeben seien fünf Prozesse auf einem Einprozessorsystem mit den angegebenen Ankunftszeiten (0 = Start) und Burst-Zeiten. Vervollständigen Sie die untenstehenden Scheduling-Pläne für die Strategie PSJF sowie die Strategie RR für die Zeitscheibenlänge 3 Zeiteinheiten. Ein Kasten im Zeitplan stellt eine Zeiteinheit dar.

4 pt

*Consider five processes on a uniprocessor system, with given arrival times (0 = start) and burst times. Complete the scheduling plans given below, for the policy PSJF and the policy RR for a timeslice length of 3 units of time. A box in the scheduling plan represents one unit of time.*

Process	Arrival Time	Burst-Time
1	1.5	3
2	0	4
3	3	5
4	2.5	2
5	5	7

### Lösung:

*PSJF*

2	2	2	2	4	4	1	1	1	3	3	3	3	3	5	5	5	5	5	5				
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--	--	--	--

*RR (timeslice length 3) – if you first requeue the running process at the end of the time slice*

2	2	2	1	1	1	4	4	2	3	3	3	5	5	5	3	3	5	5	5	5				
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--	--	--	--

*RR (timeslice length 3) – if you first enqueue newly arrived processes at the end of the time slice*

2	2	2	1	1	1	4	4	3	3	3	2	5	5	5	3	3	5	5	5	5				
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--	--	--	--

*We accept both versions.*

**-0.5 P** for each incorrect scheduling decision

**-1 P** for idle time after finished processes

- c) Berechnen Sie für den obigen PSJF-Scheduling-Plan die Wartezeit aller Prozesse. **2.5 pt**  
*For the above PSJF scheduling plan, calculate the waiting time of each process.*

**Lösung:**

*The waiting time for each process can be calculated as  $t_{wait} = t_{finish} - t_{arrival} - t_{burst}$*

Process	Finish time	Arrival time	Burst time	Waiting time
1	9	1.5	3	4.5
2	4	0	4	0
3	14	3	5	6
4	6	2.5	2	1.5
5	21	5	7	9

- d) Berechnen Sie für den obigen RR-Scheduling-Plan die Tournaround-Zeit aller Prozesse. **2.5 pt**  
*For the above RR scheduling plan, calculate the tournaround time of each process.*

**Lösung:**

*The turnaround time for each process can be calculated as  $t_{tournaround} = t_{finish} - t_{arrival}$*

Process	Finish time	Arrival time	Turnaround time
1	6	1.5	4.5
2	9 / 12	0	9 / 12
3	17	3	14
4	8	2.5	5.5
5	21	5	16

*(for first requeueing the running process / first enqueueing newly arrived processes at the end of the time slice. We accept both versions if the solution is consistent with question 2b.)*

- e) Wie verhält sich ein RR-Scheduler im Fall einer unendlich langen Zeitscheibe? **1 pt**  
*How does a RR-Scheduler behave in case of an infinite timeslice length?*

**Lösung:**

*Round-Robin selects processes in the order they arrive, and, for an infinite timeslice length, does not preempt processes until they finish. Hence, a Round-Robin-scheduler with an infinite timeslice length behaves exactly like a first-come-first-served (FCFS) scheduler.*

*(either naming or describing FCFS is sufficient)*

- f) Erklären Sie, warum ein *RR-Scheduler* keine gute Wahl für ein Batch-System ist, auf dem ausschließlich langlebige Prozesse laufen.

**2 pt**

*Explain why a RR-scheduler is not a good choice for a batch-system executing only long-running processes.*

**Lösung:**

*Round-robin schedulers emphasize interactivity by constantly switching between runnable processes. In a batch system, however, interactivity is not required. (1 P) In addition, a round-robin scheduler will cause many context switches throughout the lifetime of each process, while other schedulers like FCFS or SFJ switch processes only on completion. Since context switches are not free in a realistic system, a round-robin scheduler will likely cause much more overhead than FCFS/SJF. (1 P)*

**Total:  
13.0pt**

### Aufgabe 3: Koordination von Prozessen / Assignment 3: Process Coordination

Betrachten Sie das Code-Beispiel in Listing 2. Jeweils ein Thread führe die Funktion `thread1()` bzw. `thread2()` aus.

Consider the code example in Listing 2. Each of the functions `thread1()` and `thread2()` is executing in its own thread.

Listing 2

```

int l, m;
void thread1() {
    int key;
    do {
X1:    wait(m);
        key = 1;
X2:    while(key) swap(&l, &key);
        long_activity(); // takes 5ms
X3:    l = 0;
X4:    signal(m);
    } while (1);
}

void thread2() {
    int key;
    do {
        key = 1;
Y1:    while(key) swap(&l, &key);
Y2:    wait(m);
        short_activity(); // takes 1ms
Y3:    signal(m);
Y4:    l = 0;
    } while (1);
}

```

- a) Wie heißen die Synchronisationsmechanismen, die in Listing 2 zum Einsatz kommen? **2 pt**

*How do we call the synchronization primitives used in Listing 2?*

**Lösung:**

*Bei X1/X4 und Y2/Y3: Semaphore (oder Mutex) (1 P)*

*Bei X2/X3 und Y1/Y4: Spinlock (1 P) (atomic operations (0.5 P) )*

- b) Welche Eigenschaft muss die Funktion `swap` in den Zeilen X2 und Y1 haben, damit der Synchronisationsmechanismus korrekt funktioniert? Welches Kriterium für eine Lösung des Problems kritischer Abschnitte wird dadurch gewährleistet? **2 pt**

*Which property of the function `swap` does a correct implementation of the synchronization primitive require? Which of the criteria for a solution to the critical-section problem is thereby guaranteed?*

**Lösung:**

*`swap` must be atomic (1 P) , thereby guaranteeing mutual exclusion (1 P) for the spinlock.*

- c) Welcher unerwünschte Systemzustand kann in diesem Szenario auftreten? Illustrieren Sie diesen Zustand mit einer geeigneten graphischen Repräsentation aus der Vorlesung. Verwenden Sie als Beschriftungen die Bezeichner aus Listing 2.

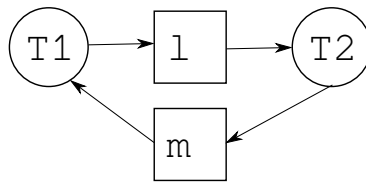
3 pt

*Which undesired system state could arise in this scenario? Illustrate that state using an appropriate graphical representation from the lecture. Use the identifiers from Listing 2 as labels.*

Undesired system state: Illustration:

**Lösung:**

Deadlock (1 P) ; Vertices of RAG (1 P) ; Edges of RAG (1 P)



- d) Wie könnten Sie das Problem durch Ändern der mit Labels markierten Zeilen (X1-4 und Y1-4) in Listing 2 lösen? Entfernen Sie keinen Code. Welchem allgemeinen Konzept zur Lösung dieses Problems entspricht Ihr Vorgehen? Nennen Sie sowohl das Konzept als auch die übergeordnete Kategorie.

3 pt

*How could you solve this problem by modifying the lines with labels (X1-4 and Y1-4) in Listing 2? Do not remove code. To which concept for solving this problem does your modification relate to? Name both the concept and the more general category.*

**Lösung:**

Exchange lines Y1 and Y2, and Y3 and Y4 (1 P) . This modification implements resource ordering (1 P) , which is one technique for deadlock prevention (1 P) .

- e) Im Code-Beispiel in Listing 2 schützen offenbar zwei Synchronisationsmechanismen einen kritischen Abschnitt. Im hier beschriebenen Szenario ist einer davon unnötig. Welchen der beiden Mechanismen würden Sie auf einem Einprozessorsystem bevorzugen? Erläutern Sie warum.

2 pt

*In the example in Listing 2 apparently two synchronization primitives protect one critical section. In the scenario we describe here, one of them is unnecessary. Explain why you would prefer one of them over the other on a uniprocessor system.*

**Lösung:**

We prefer the semaphore over the spinlock (0.5 P) , as it does not use busy wait (0.5 P) . Instead, the semaphore blocks the waiting thread, so that other threads can make progress. On a uniprocessor system, the lockholder thread cannot make progress and release the lock while another thread is busy-waiting for the spinlock (1 P) .

**Total:  
12.0pt**

## Aufgabe 4: Speicher / Assignment 4: Memory

Betrachten Sie ein System, das mittels einer zweistufigen, hierarchischen Seitentabelle virtuelle in physische Speicheradressen übersetzt. Jede Stufe besteht aus 1024 Einträgen. Sowohl der virtuelle als auch der physische Adressraum sind 4 GiB groß, wobei jede Seite 4096 Bytes umfasst. Die Wortbreite des Prozessors beträgt 4 Bytes.

*Consider a system that translates virtual addresses to physical addresses using two-level, hierarchical page tables. Every level comprises 1024 entries. The size of both the virtual and the physical address spaces is 4 GiB. The page size is 4096 bytes. The word length of the CPU is 4 bytes.*

- a) In welche Teile wird eine virtuelle Adresse im gegebenen System für die Adressübersetzung zerlegt? Nennen Sie jeweils die Breite in Bits. **2 pt**

*Into which parts is a virtual address in the given system split to perform address translation. For each part give its length in bits.*

### Lösung:

*Page directory index, page table index and byte offset (1 Pt).*

*4 GiB address space  $\Rightarrow$  32 bit addresses. The lengths are: 10 bits (PD), 10 bits (PT), 12 bits (Offset) (1 Pt).*

- b) Erklären Sie kurz, was man unter lokaler und globaler Seitenersetzung versteht. **2 pt**

*Shortly explain the terms local and global page replacement.*

**Local:** *Only frames of the faulting process are considered for replacement (1 Pt).*

**Global:** *All frames are considered for replacement; one process can take a frame from another (1 Pt).*

- c) Wie viele physische Seiten werden mit Pre-Paging maximal benötigt um einen zusammenhängenden Puffer der Länge 4 MiB in einen bereits laufenden Benutzerprozess einzublenden? Gehen Sie davon aus, dass der Puffer an Wortgrenzen, nicht jedoch an Seitengrenzen ausgerichtet ist. **2 pt**

*How many physical pages are required in the worst case with pre-paging to map a 4 MiB contiguous memory buffer into a running user process? Assume that the buffer is word-, but not page-aligned.*

### Lösung:

*4 MiB buffer and 4 KiB pages  $\Rightarrow$  1024 frames to back the buffer (0.5 Pt).*

*The buffer is not page-aligned. We therefore need another frame to accommodate for overlap. Additionally, to make the buffer visible in a user process, we require two extra frames for the page tables. A frame for the page directory is not needed because the process is already running (1.5 Pt).*

*Total:  $1024 + 1 + 2 = 1027$  frames.*

- d) Wie viele Speicherzugriffe muss die CPU maximal durchführen um diesen Puffer vollständig zu lesen, wenn weder TLB noch Cache vorhanden sind? Jeder Zugriff umfasst ein Wort.

2 pt

*How many memory accesses does the CPU have to perform in the worst case to read this entire buffer without a TLB or cache, reading one word at a time?*

**Lösung:**

*4 MiB buffer and 4 byte words  $\Rightarrow$  1 M accesses to read the buffer (1 Pt).*

*For each access the page directory entry (PDE) and the page table entry (PTE) need to be read for address translation (1 Pt).*

*Total:  $1M + (2 * 1M)$*

*Also tolerated:  $4M + (2 * 4M)$  if read address is incremented only one byte (not word) at a time.*

- e) Würde der Einsatz eines TLB die Anzahl der benötigten Speicherzugriffe reduzieren? Begründen Sie Ihre Antwort.

1 pt

*Would a TLB reduce the number of required memory accesses? Explain your answer.*

**Lösung:**

*Yes (0.5 Pt). With a TLB, only the first access to a page would require a lookup in the page table hierarchy (0.5 Pt).*

- f) Bewerten Sie folgende Aussage: Speicherallokation mit dem Buddy-System ist frei von Fragmentierung. Begründen Sie Ihre Antwort.

1 pt

*Evaluate the following statement: Memory allocation with the buddy-system is free of fragmentation. Explain your answer.*

**Lösung:**

*The statement is wrong (0.5 Pt). The buddy system suffers from internal and external fragmentation (0.5 Pt).*

- g) Für jede CPU in einem Mehrprozessorsystem enthält die Struktur `MyStruct` einen Zähler, den die entsprechende CPU häufig aktualisiert. Obwohl die CPUs auf getrennten Zählern arbeiten, ist die Performance schlecht. Woran liegt das? Welche grundsätzliche Modifikation an `MyStruct` schlagen Sie vor?

**2 pt**

*For each CPU of a multi-processor-system the structure `MyStruct` contains a counter, which the respective CPU updates frequently. The CPUs access different variables, though the performance is poor. Why? What general modification to `MyStruct` can you propose?*

```
struct MyStruct {  
    int counterForCpu1;  
    int counterForCpu2;  
};
```

**Lösung:**

*The counters are placed on the same cache line. That leads to a performance hit due to frequent mutual cache line invalidation on updates (**1 Pt**).*

*A solution is to add padding to each counter to force them onto different cache lines (**1 Pt**).*

*Also tolerated: Use separate structures for each counter and pointers. Note that in general this does not guarantee that the memory allocator (e.g., `malloc`) positions the counters at addresses mapping to different cache lines!*

**Total:  
12.0pt**

## Aufgabe 5: Hintergrundspeicher / Assignment 5: Secondary Storage

- a) Nennen Sie die drei grundlegenden I/O-Techniken. Welche Technik wird für Festplatten und SSDs verwendet und warum?

2 pt

*Name the three basic I/O techniques. Which one is used for harddisks and SSDs, and why?*

### Lösung:

*Programmed I/O / Polling (0.5 P) Interrupt-driven I/O (0.5 P) Direct Memory Access (DMA) (0.5 P) DMA is typically used by SSDs and HDDs, as it allows to transfer data at high throughput with low CPU utilization.(0.5 P)*

- b) Für Solid State Disks (SSDs) wurde ein neuer Befehl in die Schnittstellen zu Hintergrundspeichern eingeführt. Wie lautet dieser Befehl und wann sollte ein Dateisystem ihn ausführen? Welche zwei Vorteile bietet dieser Befehl aus Sicht der SSD?

3 pt

*For Solid State Disks (SSDs) a new command has been introduced to the background storage interface. What is its name and when is it executed? Name two advantages of this command from the perspective of an SSD!*

### Lösung:

*trim (0.5 P) should be called whenever any disk blocks are freed, that is, when a file is deleted or truncated (0.5 P) .*

*Unused logical blocks do not have to be copied in case of a SSD block re-write (avoids Write Amplification) (1 P)*

*Unused logical blocks increase the pool of spare SSD blocks (0.5 P) after garbage collection (0.5 P)*

- c) Welche der folgenden Informationen findet sich üblicherweise (a) in der inode, (b) im Verzeichnis, oder (c) in der Datei? (Pro Zeile ist genau eine Antwort richtig)

2 pt

*Which information is typically stored in (a) the inode, (b) the directory, or (c) the file? (In each line, there is exactly one correct answer)*

Inode/ <i>Inode</i>	Verzeichnis/ <i>Directory</i>	Datei/ <i>File</i>	
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Dateiname <i>file name</i>
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Anzahl der Hardlinks <i>number of hard links</i>
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Magische Zahl für den Dateityp <i>magic number for the file type</i>
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Zugriffsrechte <i>access rights</i>

- d) Gegeben sei ein Dateisystem, das die Belegung von Blöcken mit einer verketteten Liste im Speicher verwaltet. Der aktuelle Zustand dieser Liste auf der Festplatte sei in Tabelle 1 angegeben. Ein Verzeichnis enthält die Einträge {"foo":3, "bar":7}. Welche Blöcke enthalten die Inhalte der Dateien *foo* und *bar*?

2 pt

*Imagine a file system with linked list allocation in memory. The current state of that list on disk is given below in Table 1. The current directory contains the entries {"foo":3, "bar":7}. Which disk blocks contain the contents of the files *foo* and *bar*, respectively?*

Table 1

Block	Link
0	1
1	5
2	
3	1
4	-1
5	4
6	
7	-1

**Lösung:**

*foo: 3, 1, 5, 4 (1 P) bar: 7 (1 P)*

- e) Ein weiteres Verzeichnis im oben beschriebenen Dateisystem enthält einen Eintrag "punktliste.txt":0. Was ist daran problematisch? Wie kann der Einsatz eines Buffer Cache zu dieser Situation führen?

**2 pt**

*Another directory in the file system described above contains an entry "punktliste.txt":0. What is the issue with that entry? How can the use of a buffer cache lead to that situation?*

**Lösung:**

*The file punktliste.txt appears to contain some of the blocks of file foo, which is an inconsistent file system state (1 P) .*

*Modifications to the file system get written to the buffer cache first. Apparently, the entry for block 0 in the linked allocation list has been changed in memory but not yet on disk (1 P) .*

*When the system crashes (e.g., power failure) before all the changes are written back to disk, the file system can be in an inconsistent state when it is mounted during a reboot (alternative answer).*

**Total:  
11.0pt**