

Aufgabe/Question 1 (Zum Aufwärmen/Warm up, 3 + 3 + 1 + ... + 1 Punkte/marks)

1. „Zählen Sie drei Vorteile von **verteilten Systemen** gegenüber **zentralen Systemen** auf und geben Sie **jeweils** hierfür eine **knappe Begründung!**“
*“Enumerate three advantages of **distributed systems** over **centralized ones** and add a **short justification** for **each** of them.”*
 - a) Reliability, if one node crashes the rest of the system still works
 - b) Scalability, you can add nodes incrementally
 - c) Economics, better performance/prices ratio

2. „Was verstehen Sie unter dem Begriff **„Dienstgüte eines Systems“** (*quality of service*) und durch welche **Aktionen** kann diese Dienstgüte **gefährdet** werden?“
*“What does the term **“quality of service”** mean and by what **actions** can this quality of service be **threatened**?”*

Quality of Service includes availability and predictable performance, e.g. if a client requested for a certain bandwidth, the system has to guarantee that he will get it.

Without a load control users can flood a system (see amazon.com) with dummy requests.

Einige der folgenden Aussagen sind korrekt, einige inkorrekt. Unterstreichen Sie „korrekt“, wenn die Aussage korrekt ist, unterstreichen Sie „inkorrekt“, wenn die Aussage inkorrekt ist!
Some of the following statements are correct, some are incorrect. Underline “korrekt” if the statement is correct; underline “inkorrekt” if the statement is incorrect.

3. „**READ_DISKBLOCK(...)** und **WRITE_DISKBLOCK(...)** sind typische Schnittstellenoperationen eines Mikrokerns der 2. Generation.“
*“**READ_DISKBLOCK(...)** and **WRITE_DISKBLOCK(...)** are typical interface-operations of a “second generation” micro kernel .”*

korrekt

inkorrekt

4. „Der **wesentliche Beitrag** des „Mach“-Mikrokerns im Hinblick auf die Systemarchitekturforschung bestand in dessen **schnellen Interprozesskommunikationen** (fast IPC's).“
“*The essential contribution of the “Mach” micro-kernel to the research field of system architecture was its fast inter process communications (fast IPC’s).*”

korrekt

[inkorrekt](#)

5. „Der **Linux-Kern** ist Repräsentant einer **monolithischen** Systemarchitektur.“
“*The Linux kernel represents a monolithic system architecture.*”

[korrekt](#)

inkorrekt

6. „Jeder **Leseauftrag** READ_FILE(...) an einen **zustandslosen** Dateidienst (*file server*) muß eine **Dateiidentifikation** als **Parameter** mit beinhalten.“
“*Each read request READ_FILE(...) to a stateless file server has to include a file identifier as a parameter.*”

[korrekt](#)

inkorrekt

7. „Im **Überlastfall** ist die Plattenzugriffsstrategie „**Minimale Suchzeit**“ (*minimal seek-time first*) **fair**.“
“*In case of an overload situation, the policy “minimal seek-time first is fair.*”

korrekt

[inkorrekt](#)

8. „Ein **Seitenfehler** (page fault) ist eine **Ausnahme** (exception) und keine **Unterbrechung** (interrupt).“
“*A page fault is an exception, not an interrupt.*”

[korrekt](#)

inkorrekt

Aufgabe/Question 2

(2 + 3 + 2 + 2 + 3 Punkte/marks)

1. „Wie heißt die **Kerndatenstruktur**, die einen **Thread repräsentiert**. Zählen Sie mindestens **3 typische Einträge** dieser Kerndatenstruktur auf!“

”What is the term used for the **kernel data-structure representing a thread**? Enumerate at least 3 typical entries of this kernel data-structure.”

Name of kernel data-structure: **Thread-Control-Block(TCB)**

Typical entries of this kernel data-structure: **Kernel Stack-Pointer,**

Status-Register,

Thread-Priority

2. „Ein lokales System offeriert, sequentielle bzw. parallele Applikationen mittels „User“- oder „Kernel-Level“ Threads zu implementieren. Wie viele der in 2.1. erwähnten Kerndatenstrukturen zur Repräsentation der Threads gibt es jeweils bei den folgenden konkreten Beispielen?“

“A local system offers the possibilities for implementing sequential and parallel applications by user- or kernel-level threads. How many of those kernel data-structures mentioned in 2.1 for thread representation are needed in the following example applications?”

Single-threaded Application: **1**

Multi-threaded Application with 4 Kernel-Level Threads: **4**

Multi-threaded Application with 4 User-Level Threads: **[1, 4]**

3. „Im Kern des obigen Systems gibt es ein zentrales Unterprogramm (Prozedur) „THREAD_SWITCH“, das den eigentlichen Kontextwechsel zwischen den relevanten Threads durchführt. Warum wird in diesem Unterprogramm beim Wechsel von einem zum nächsten Thread der aktuelle Befehlszeiger (*instruction pointer*) nicht explizit gerettet?“

“In the kernel of the above system there is one central subroutine (procedure) “THREAD_SWITCH.” which does the context switch between the relevant threads. Why do we not save the current instruction pointer explicitly within this subroutine when switching from one thread to the next one?”

Within the procedure THREAD_SWITCH you are exchanging the 2 stacks at exactly the same instruction address whenever you are performing this procedure, i.e. you are always performing the next instruction of THREAD_SWITCH on behalf of the new thread, you just have exchanged the surrounding frame-box of the caller.

Fortsetzung von Aufgabe 2 / Question 2 continued:**(2 + 3 Punkte/marks)**

4. „Beschreiben Sie so knapp wie möglich, gleichwohl so ausführlich wie nötig, worin sich die **Threadzustände** „**Rechnend**“ (*running*), „**Bereit**“ (*ready*) und „**Blockiert**“ (*blocked*) in einem Mehrprozessorsystem von einander unterscheiden.“
*“Describe as short as possible, but as thorough as necessary how the **thread-states** “**Running**”, “**Ready**”, and “**Blocked**” differ from each other within a multi-processor system.”*

Only the running threads currently have a CPU to be executed on.

However, because at least 2 CPUs may try to assign a new ready thread we have to avoid that 2 CPUs try to assign the same thread, i.e. we have to distinguish between running and ready threads.

Ready threads are those that currently just waits for a free CPU, in contrast to blocked threads which are waiting for a specific additional event, e.g. “waiting for the end of a disk-I/O” or “waiting for an incoming message” etc.

5. „Beschreiben Sie **Vor- und Nachteile** einer **zentralen Bereitwarteschlange** gegenüber **lokalen Bereitwarteschlangen pro Prozessor** in einem Mehrprozessorsystem!“
*“Describe the **pros and cons** of **one central ready-queue** in contrast to **local ready-queues per processor** within a multi-processor system.”*

Advantage of a central ready queue:

You can easily establish a specific global order within all ready threads reflecting any specific global policy that has to be supported by all CPUs. The danger of idle CPUs even though there are still ready threads is minimized.

Disadvantage:

It's more difficult to support processor affinity. The more CPUs the more this central queue might become a bottleneck.

Advantage of local queues:

It's easy to implement the processor affinity model, i.e. you may want to preserve threads on the same CPU forever, using their old cache-working sets.

Disadvantage:

Load-balancing problem, no true global priority schema.

Fortsetzung von Aufgabe 3 / Question 3 continued: (4 + 2 Punkte/marks)

2. „Vergleichen Sie die **Güte** des **token-basierten Verfahrens** aus 3.1 mit der des **zentralisierten Verfahrens** hinsichtlich der **relevanten Leistungsmaße!**“
 “Compare the *quality* of the *token-based algorithm* of 3.1 with a *centralized algorithm* with respect to the *relevant performance measures*.”

	Centralized Algorithm	Token-Algorithm
# Messages per CS	3	0 ... ∞
Delay Time	2 T	N *T/2
Response Time	2T + E	N*T/2 + E
Problems	crash of central node	node crashes loss of token

T = Message Transfer Time

E = Execution Time of CS

N = # competing processes

3. „Welches sind die **beiden entscheidenden Nachteile** eines **zentralisierten Verfahrens?**“
 “What are *the two major drawbacks* of a *centralized algorithm*?”

Single Point of Failure \Rightarrow Total Loss of Functionality

Central Node a potential Bottleneck \Rightarrow No Quality of service

Aufgabe/Question 4**(4 + 2 + 3 + 3 Punkte/marks)**

1. „Beschreiben Sie so knapp wie möglich, gleichwohl so ausführlich wie nötig den **reinen LRU-Seitentauschalgorithmus (paging algorithm)** auf einem Beiblatt! (Eine Skizze könnte zwecks Zeitersparnis und zum besseren Verständnis hilfreich sein)“
 “Describe as short as possible, but as thorough as necessary, the **pure LRU-paging algorithm** on one of the draft pages. (Drawing a figure may save time and help understanding.)”

Take the page that hasn't been referenced for the longest period of time.

2. „**Begründen** Sie, warum der **reine LRU-Seitentauschalgorithmus** in der **Praxis nicht** verwendet wird.“
 “**Explain**, why the **pure LRU-paging algorithm** is **not** used in **practice**.”

Either you have to update your LRU-stack per new page number in the page reference string (extremely time consuming) or the hardware offers per page frame an entry for the reference-time (expensive) and in case of a page-in you have to find out the page with the minimal-time stamp

3. „Gegeben sei ein **Hauptspeicher** mit 3 Kacheln (*page-frames*), die anfangs leer seien. Ergänzen Sie in der unten angegebenen Speicherskizze die Belegung der Kacheln durch entsprechende Seitennummer, wenn für den angegebenen **Seitenreferenzstring** als Seitentauscher der Clockalgorithmus eingesetzt wird! Annahme: Initial zeige der Clockzeiger auf Kachel 0.“
 “There is a **main memory** of just 3 page-frames which are initially empty. In the table below fill out the page-frames with the corresponding page-numbers according of the page reference string. Assume that the clock-algorithm does paging. Initially the clock hand points to page-frame 0.”

Refrenzstring	1	2	1	3	1	2	1	4	1	2	3	4
Kachel 0	1/1	1/1	1/1	1/1	1/1	1/1	1/1	4/1	4/1	4/1	3/1	3/1
Kachel 1		2/1	2/1	2/1	2/1	2/1	2/1	2/0	1/1	1/1	1/0	4/1
Kachel 2				3/1	3/1	3/1	3/1	3/0	3/0	2/1	2/0	2/0

X = Pagefault due to page X

X/1 = reference bit of page frame mapped to page X is set,

X/0 = reference bit of page frame mapped to page X is cleared

Fortsetzung von Aufgabe 4 / Question 4 continued:**(3 Punkte/marks)**

4. „Annahme: Die eingesetzte **MMU-Hardware** unterstützt die drei **Kontrollbits: Referenz-Bit, Modified-Bit** und **Valid-Bit**. Welche der folgenden **Kontrollbits** werden entweder von der MMU-Hardware direkt oder von der **Seitentauscher-Software** (*paging software*) (inklusive der Seitenfehlerbehandlung) des Clock-Algorithmus gelesen bzw. gesetzt? Kreuzen Sie in unten angegebener Tabelle die Zellen entsprechend an!“

“Assume: The involved **MMU-Hardware** supports the following **control-bits: Reference-Bit, Modified-Bit, Valid-Bit**. Which of the following control bits are read or set either by the MMU-hardware directly or by the **paging-software** (including the page-fault handling routine) of the clock-algorithm? Mark the corresponding cells of the following table correspondingly.”

Component type	MMU-Hardware	Clock-Paging Software
Action on Control-Bit		
Read Reference Bit		X
Reference Bit = 0		X
Reference Bit = 1	X	
Read Dirty Bit		X
Dirty Bit = 0		X
Dirty Bit = 1	X	
Read Valid Bit	X	
Valid Bit = 0		X
Valid Bit = 1		X

Legende:

Reference-Bit; Reference-Bit = 0, bedeutet, dass auf die entsprechende Seite in jüngster Vergangenheit nicht mehr zugegriffen worden ist. Reference-Bit = 1, bedeutet, dass auf die Seite lesend oder schreibend zugegriffen worden ist.

Modified-Bit; mit entsprechender Semantik für schreibende Zugriffe.

Valid-Bit; Valid-Bit = 1 bedeutet, dass die entsprechende Seite derzeit auf eine physische Hauptspeicherkachel abgebildet ist, Valid-Bit = 0 bedeutet, dass die entsprechende Seite derzeit nicht auf eine physische Hauptspeicherkachel abgebildet ist.

Reference-Bit; Reference-Bit = 0 means that the corresponding page has not been accessed in the recent past. Reference-Bit =1 means that the page has been read or written in the recent past.

Modified-Bit; with a corresponding meaning for write-accesses.

Valid-Bit; Valid-Bit = 1 means that the corresponding page is currently mapped to a physical page-frame of main memory, Valid-Bit = 0 means that the corresponding page is currently not mapped to a physical page-frame of main-memory.

Aufgabe 5/ Question 5:**(4 + 3 + 5 Punkte/marks)**

1. „Beschreiben Sie das **Bell-LaPadula Sicherheitsmodell!** (Eine Skizze kann wiederum sehr hilfreich sein.)“

*Describe the **Bell-LaPadula model.** (Again, drawing a figure may be very helpful.)*

The Bell-LaPadula model enables “disclosure”. Subjects of higher security levels are not allowed to write down, i.e. into objects of lower security levels. Thus you can prevent that subjects of lower levels may read confidential documents, i.e. no secret information may flow to less privileged subjects.

But, it enables blind writes, too, i.e. low level subjects may write to high level documents thus enabling an attack on data integrity.

The Bell LaPadula security model is one of several mandatory security policies.

2. „Zählen Sie mindestens drei verschiedene Implementierungstechniken „**Capabilities**“ zu schützen!“

*“Enumerate at least three different implementation techniques of **protecting capabilities** in a **protected way.**”*

Splitted Segments

Tagged Capabilities

Special containers in the OS

Encrypted Capabilities with the User's AS

Fortsetzung von Aufgabe 5 / Question 5 continued:**(5 Punkte/marks)**

3. „Als **Sicherheitsmanager** sollen Sie die **Zugangskontrolle** zum **Zentralrechner** Ihrer Firma besonders sicher machen. Entwerfen Sie hierfür ein geeignetes **Sicherheitskonzept**, wobei Sie ein **nahezu unbeschränktes Budget** zur Verfügung haben! Ihr Sicherheitskonzept soll hierbei sowohl **Schutzmassnahmen** im Umfeld des Rechners umfassen, als auch Maßnahmen, welche die **Authentifizierung** (*authentication*) eines potentiellen Benutzers kontrollieren.“

*”As the **security manager** you have to establish the **entrance-control** for the **central computer** of your company such that it is as safe as possible. Design an applicable security-concept whereby you have a **nearly unlimited budget** for implementing your concept. Your security concept should contain security controls already needed around your computer installation as well as measures, controlling the authentication of a potential user.”*

Physical Entrance Control:

Video monitoring, chip cards, biomedical control devices (finger print, retina scan, voice...)

Authentication:

Hidden password file, reject guessable passwords, insist on long and periodically altered passwords, demand additional personal dates, no remote login as administrator, monitoring user profiles

Communication and Data Storing

Use encryption in any case, install firewall,